University of Alberta

A Data Driven Subspace Approach to Multivariate Controller Synthesis and Performance Analysis

by

Ramesh Kumar Kadali

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Process Control

Department of *Chemical and Materials Engineering*

Edmonton, Alberta

Spring 2003

National Library
of Canada

Acquisitions and
Bibliographic Services

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale
du Canada

Acquisisitons et
services bibliographiques

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 0-612-82122-6
Our file Notre référence
ISBN: 0-612-82122-6

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou aturement reproduits sans son autorisation.

Canada

University of Alberta

Library Release Form

Name of Author: *Ramesh Kumar Kadali*

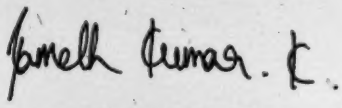
Title of Thesis: *A Data Driven Subspace Approach to Multivariate Controller Synthesis and Performance Analysis.*

Degree: *Doctor of Philosophy*

Year this Degree Granted: *2003*

Permission is hereby granted to the University of Alberta Library to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only.

The author reserves all other publication and other rights in association with the copyright in the thesis, and except as herein before provided, neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatever without the author's prior written permission.



Ramesh K. Kadali

5-5-135/5, Khammam
Andhra Pradesh – 507001
INDIA.

April 11, 2003

University of Alberta

Faculty of Graduate Studies and Research

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled **A Data Driven Subspace Approach to Multivariate Controller Synthesis and Performance Analysis** submitted by **Ramesh Kumar Kadali** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy** in **Process Control.**

.....
Biao Huang (Research Supervisor)
(Dept. of Chemical and Materials Engineering)

.....
Fraser Forbes
(Dept. of Chemical and Materials Engineering)

.....
Scott Meadows
(Dept. of Chemical and Materials Engineering)

.....
Sirish Shah
(Dept. of Chemical and Materials Engineering)

.....
Qing Zhao
(Dept. of Electrical and Computer Engineering)

.....
Ali Cinar (External Examiner)
(Dept. of Chemical and Environmental Engineering)
(Illinois Institute of Technology, Chicago, IL, USA)

Date: Apr 11, 2003

Contradictions do not exist. Whenever you think that you are facing a contradiction, check your premises. You will find that one of them is wrong.

- Ayn Rand

Dedicated To

MY PARENTS

Abstract

Subspace identification methods identify certain matrices, which capture the correlations between the process inputs and outputs in non-parametric coefficients form, as a first step by regressing the process input-output data. The focus of the research presented in this thesis is to use these subspace matrices for multivariate controller synthesis and performance analysis. The subspace matrices based approach is used for the closed loop estimation of the dynamic matrix and noise model; predictive controller design without identifying parametric models; controller performance analysis using the Linear Quadratic Gaussian (LQG) controller as a benchmark; estimation of multivariate Minimum Variance Control (MVC) benchmark without calculating the interactor matrix; and feedforward controller performance analysis for identification of the important disturbance variables. The approach used in this thesis can be considered a 'data driven approach' in the sense that no traditional parametric models are used. Hence the intermediate subspace matrices, which are obtained directly from the process data and otherwise identified as a first step in the subspace identification methods, are directly used for closed loop identification, controller design and performance analysis.

Acknowledgments

I don't want to achieve immortality through my work... I want to achieve it through not dying.

- Woody Allen

One of the consistent motivating factors during my Ph.D. program at the University of Alberta has always been my research supervisor Dr.Biao Huang. Dr.Huang has always been supportive and encouraging. As Bart Simpson learns... A burp in a bottle is not a science project... neither is it a thesis. I would like to thank Biao for guiding me in molding this thesis into a work worth the time and hard effort I put into it for a Ph.D. degree.

Thanks to my parents and siblings in India for always being supportive and encouraging. Thanks also to the many teachers from the kindergarten to the universities I attended, for all the efforts in imparting education in me.

The Computer Process Control group has always been resourceful and filled with some of the nicest people I ever come across. The process control courses taught by Dr.Shah and Dr.Forbes were helpful in acquiring the necessary background for research in this area. The graduate students in CPC are ever available for discussing ideas and trying new things. I couldn't have been better accommodated in the group. Thanks are due to Dr.Huang's research group for all the support during the thesis work. Thanks are also due to NSERC and Syncrude Canada Ltd. for providing the funding for my Ph.D. work, and the process control personnel at the Extraction and Upgrading departments of Syncrude for all the support during the industrial work

for the thesis.

The few years I spent at the Univ. of Alberta campus in Edmonton have been quite enjoyable thanks to the people I had been friends with or had the pleasure of sharing ideas with. All of them are precious moments and cherished memories. Thanks to Ram Maikala for thoughtful advises and lending a helpful hand on occasion. Ashish, it was fun sharing the office with you. Thanks to Bhushan, Hari, Arun, Xin, Raghu, Rohit, Misha, Donguang, Dr.Hu and others. You are all true examples of friends enriching life.

Last but not the least, thanks are due to Brian Doucette and Brad Bjornson at Suncor Energy Inc. for providing me an opportunity to work in such a nice work environment.

Ramesh Kadali

April 11, 2003.

Contents

1 Introduction **1**

1.1 An overview of objectives of this thesis 1

1.2 Contributions of this thesis 4

1.2.1 Contributions to the theory 4

1.2.2 Contributions via practical applications 6

1.3 Organization of the thesis 6

2 Review on subspace identification methods **9**

2.1 Introduction 9

2.2 Subspace matrices description 10

2.3 Regression techniques 13

2.4 Open loop subspace identification methods 14

2.4.1 N4SID 15

2.4.2 MOESP 17

2.4.3 CVA 17

2.5 Closed loop subspace identification methods 18

2.5.1 N4SID approach 18

2.5.2 MOESP/ CVA approaches 20

2.5.3 Ljung and McKelvey's approach via estimated predictors . . . 21

3 Estimation of Dynamic matrix and noise model for MPC using closed loop data 24

3.1 Introduction 24

3.2 Estimation of the process dynamic matrix and the noise model from closed loop data 28

3.2.1 Estimation of the process model 32

3.2.2 Estimation of the noise model 33

3.3 The algorithm 34

3.4 Some guidelines for the practical implementation of the algorithm . . 35

3.5 Extension to the case of measured disturbance variables 37

3.6 Closed loop simulations 38

3.6.1 Univariate system 39

3.6.2 Multivariate system 40

3.7 Identification of the dynamic matrix: Practical application on a pilot scale plant 41

3.8 Conclusions 42

4 A data driven subspace approach to predictive controller design 50

4.1 Introduction 50

4.2 Revisit of GPC 53

4.3 Predictive controller design from subspace matrices 54

4.3.1 Inclusion of integral action through integrated noise model . . 56

4.3.2 Inclusion of feedforward control 59

4.3.3 Constraint handling 60

4.4 Tuning the noise model 61

4.5 Simulations 64

4.6 Experiment on a pilot scale process 66

4.7 Conclusions 67

5 Controller performance analysis with LQG-benchmark obtained under closed loop conditions 76

5.1 Introduction 76

5.2 Designing LQG-controller using subspace matrices 80

5.3 Obtaining LQG benchmark from closed loop data 83

5.4 Profit analysis of feedforward control 86

5.5 Controller performance analysis 89

5.5.1 Case 1: Feedback controller acting on the process with no measured disturbances 90

5.5.2 Case 2: Feedforward plus feedback controller acting on the process 91

5.5.3 Case 3: Feedback controller acting on the process with measured disturbances 92

5.6 Summary of the subspace matrices approach to the calculation of LQG-benchmark 93

5.7 Simulations 94

5.8 Application on a pilot scale process 96

5.9 Conclusions 97

6 Estimation of the statistical confidence intervals for LQG-benchmark obtained using the subspace matrices 103

6.1 Introduction 103

6.2 Kronecker algebra and matrix calculus 104

6.2.1 Kronecker algebra 104

6.2.2 Differentiating a matrix or a matrix function w.r.t. a matrix . 105

6.3 Statistical analysis for matrices 105

6.3.1 Taylor series expansion 106

6.3.2 Basic derivations 106

6.4 Statistical properties of matrices as random variables 108

6.5 Multivariate multiple regression 113

6.6 Statistical properties for subspace matrices 115

6.6.1 Subspace matrices identification from the open-loop process data 115

6.6.2 Subspace matrices identification from the closed-loop process data 116

6.7 Statistical properties for the estimated LQG benchmark coefficients . 119

6.8 Simulations 122

6.9 Conclusions 123

7 Performance assessment of multivariate feedback and feedforward controllers without interactor matrix **124**

7.1 Introduction 124

7.2 Design of minimum variance control using subspace matrices 128

7.3 Estimation of the Multivariate MVC-benchmark 129

7.4 Equivalence of subspace approach to the conventional transfer function approach in obtaining the MVC-benchmark variance 135

7.4.1 Univariate system 142

7.5 Multivariate minimum variance benchmark for the feedforward plus feedback control case 143

7.5.1 feedforward plus feedback minimum variance controller 143

7.5.2 MVC-benchmark 144

7.6 Algorithm for obtaining the MVC benchmark variance from routine operating data for multivariate systems 146

7.7 Numerical example 147

7.8 Conclusions 149

8 Dynamic multivariate analysis of variance **150**

8.1 Introduction 150

8.2 Analysis of variance for feedforward control 151

8.3 The subspace matrices approach 153

8.4 Simulations 156

8.5 Conclusions 157

9 Synthesis and analysis of an industrial MPC application **158**

9.1 Introduction 158

9.2 Process description 159

9.3 Identification 161

9.4 Tuning and Optimization 162

9.5 Performance assessment 162

9.5.1 Online predictions validation 163

9.5.2 Primary assessment of controller performance 163

9.5.3 Controller optimizer performance assessment 164

9.5.4 Closed loop settling time based performance assessment 165

9.5.5 MVC-benchmark calculation using subspace approach with weighting matrices 166

9.5.6 LQG-benchmark calculation using subspace approach with weighting matrices 168

9.6 Conclusions 171

A Closed loop subspace identification when a FF plus FB controller is acting on the process **191**

B QP formulation for constraints handling **193**

C Equivalence of subspace and GPC predictor matrices **195**

D Equivalent subspace representation **197**

E Simplified matrix expressions from singular value decomposition **199**

F Essentially disjoint condition 201

G Corollaries from [39] 203

H Scope for future work 204

List of Tables

5.1 LQG-benchmark performance assessment parameters for the simulations. 95

5.2 LQG-benchmark performance assessment parameters for the pilot scale process. 97

8.1 Contribution of the principle components to process output variance . 157

8.2 Correlation matrix table 157

9.1 Identified transfer functions between the CVs and MVs. 161

9.2 Identified transfer functions between the CVs and DVs. 162

9.3 MVC-benchmark based controller performance assessment. 168

9.4 LQG-benchmark based controller performance assessment. 171

List of Figures

1.1 Research Overview 8

2.1 Subspace representation 23

2.2 Closed loop system for subspace identification. 23

3.1 Closed loop system. 43

3.2 Comparing the existing closed loop subspace state space identification methods and the new approach. 44

3.3 univariate system: Closed loop system data 44

3.4 univariate system: Comparing the true (solid) IR-coefficients with those obtained in the subspace matrices (dotted). 45

3.5 univariate system: Comparing the true (solid) IR-coefficients with those identified by MOESP/CVA approaches. 45

3.6 Multivariate system: Closed loop system data 46

3.7 Multivariate system: Comparing the true (solid) IR-coefficients with those obtained in the subspace matrices (dotted). 46

3.8 Multivariate system: Comparing the true (solid) IR-coefficients with those identified by MOESP approach. 47

3.9 Experimental setup. 47

3.10 Pilot scale process: Closed loop system data. 48

3.11 Pilot scale process: IR-coefficients from the consecutive columns of the identified subspace matrices. 48

3.12 Pilot scale process: Comparing the IR-coefficients from subspace matrices identified using the open loop data (dotted line) and that from the closed loop data (solid line). 49

4.1 Inputs, measured disturbance and outputs data from simulations. . . 68

4.2 Comparison of the process and noise models from subspace matrices with the true models. 69

4.3 Predictive controller without and with the integrator. w(t)=0; **e(s)** =0. 70

4.4 Predictive controller without and with the feedforward control. **e(s)** =0. 70

4.5 Variation of input weighting, λ. w(t)=0; **e(s)** =0. 71

4.6 Variation of control horizon, N_u. w(t)=0; **e(s)** =0. 71

4.7 Variation of prediction horizon, N_2. w(t)=0; **e(s)** =0. 72

4.8 Constrained predictive controller. w(t)=0; **e(s)** =0. 72

4.9 Noise model tuning for model mismatch. w(t)=0; **e(s)** =0. 73

4.10 Experimental setup 73

4.11 Step response coefficients from the subspace matrices identified using the open loop data. 74

4.12 Subspace based predictive controller on the pilot scale process. 74

4.13 Subspace based predictive controller for the constrained case on the pilot scale process. 75

5.1 Optimal LQG control performance limit curve. V_u and V_y represent the variances obtained from process data while V_u^o and V_y^o represent the optimal LQG-benchmark variances. 99

5.2 Optimal LQG control performance limit curve. 99

5.3 LQG-benchmark variances of the input. 100

5.4 LQG-benchmark variances of the output. 100

5.5 Experimental setup 101

5.6 Impulse response models for the process and noise identified using closed loop data 101

5.7 Optimal LQG-benchmark curve for the CSTH 102

9.1 Settling vessel 160

9.2 Closed loop data of the CVs under MPC. 173

9.3 Comparing the model-predictions with the true values (–) of the controlled variables. 173

9.4 Auto correlation function of the prediction errors. 174

9.5 Cross correlation coefficients between $Error = (CV1_{SP} - CV1)$ and MVs. 174

9.6 Cross correlation coefficients between $Error = (CV2_{SP} - CV2)$ and MVs. 175

9.7 Cross correlation coefficients between $Error = (CV3_{SP} - CV3)$ and MVs. 175

9.8 Cross correlation coefficients between $Error = (CV4_{SP} - CV4)$ and MVs. 176

9.9 Cross correlation coefficients between $DV1$ and MVs. 176

9.10 Cross correlation coefficients between $DV2$ and MVs. 177

9.11 Optimization function operation. 177

9.12 Auto correlation function of the CVs. The vertical lines represent the maximum closed loop settling times specified in the controller design. 178

H.1 Future research scope 206

Nomenclature

Notation	Description	Section
A, B, C, D	Dynamic state space system matrices for the time-invariant system defined in equations (2.1)-(2.2).	2.2
a_k	white noise signal	4.3.1
A_c, B_c, C_c, D_c	Dynamic state space system matrices for the controller Q.	3.2
C_{lqg}	State feedback control law derived for system defined in equations (2.1)-(2.2).	5.2
C^b_{lqg}	State feedback control law derived for system defined in equations (5.33)-(5.34).	5.4
$A(z^{-1}), B(z^{-1})$ $C(z^{-1})$	Polynomials in the backshift operator, z^{-1}	4.2
D_p	$\begin{bmatrix} Y_p \\ U_p \\ R_p \\ E_p \end{bmatrix}$	3.2
e_k	System unmeasured disturbance(s) at sampling instant k.	2.2
$E[*]$	Expectancy operator	
$(E)^{fb}$, $(E)^{ff\&fb}$	LQG-benchmark based controller performance indices with respect to the process input variance.	5.5
E_p	*past* Data Hankel matrix for e_k	2.2
E_f	*future* Data Hankel matrix for e_k	2.2
E_k, F_k	Polynomials obtained in Diophantine expansions	4.2

F	Free response of the process output.	4.3.1
G	Dynamic matrix containing the step response coefficients	4.2
G_k, Γ_k	Polynomials obtained in Diophantine expansions	4.2
$G(z^{-1})$	Transfer function representation of the deterministic part of the system	4.4
h	Number of system measured disturbance(s).	4.3.2
$H(z^{-1})$	Transfer function representation of the stochastic part of the system.	4.4
H_N	Lower triangular Toeplitz matrix for the deterministic part of the system in equations (2.1)-(2.2), $\begin{bmatrix} D & 0 & \dots & 0 \\ CB & D & \dots & 0 \\ \dots & \dots & \dots & \dots \\ CA^{N-2}B & CA^{N-3}B & \dots & D \end{bmatrix}$.	2.2
H_N^c	Lower triangular Toeplitz matrix for the stochastic part of the controller in equations (3.2)-(3.3), $\begin{bmatrix} D_c & 0 & \dots & 0 \\ C_cB_c & D_c & \dots & 0 \\ \dots & \dots & \dots & \dots \\ C_cA_c^{N-2}B_c & C_cA_c^{N-3}B_c & \dots & D_c \end{bmatrix}$.	3.2
H_N^s	Lower triangular Toeplitz matrix for the stochastic part of the system in equations (2.1)-(2.2), $\begin{bmatrix} I_m & 0 & \dots & 0 \\ CK & I_m & \dots & 0 \\ \dots & \dots & \dots & \dots \\ CA^{N-2}K & CA^{N-3}K & \dots & I_m \end{bmatrix}$.	2.2

H_N^v	Lower triangular Toeplitz matrix for the measured disturbances part of the system in equations (2.1)-(2.2), $\begin{bmatrix} D^v & 0 & ... & 0 \\ CB^v & D^v & ... & 0 \\ ... & ... & ... & ... \\ CA^{N-2}B^v & CA^{N-3}B^v & ... & D^v \end{bmatrix}$.	2.2
I_n	Identity matrix of dimension $(n \times n)$	
$(I_E)^{fb}$, $(I_E)^{ff\&fb}$	LQG-benchmark based performance improvement indices with respect to the process input variance.	5.5
$(I_\eta)^{fb}$, $(I_\eta)^{ff\&fb}$	LQG-benchmark based performance improvement indices with respect to the process output variance.	5.5
j	Number of columns of the data block-Hankel matrices.	2.2
J	Optimization function.	4.2
K	Kalman filter gain matrix.	2.2
K'	Modified Kalman filter gain matrix, $K' = K + K^*$.	2.2
K_{lqg}	LQG-control gain defined in equation (5.19).	5.3
l	Number of system input(s).	2.2
L_e	Subspace matrix containing the noise model Markov parameters.	2.2
L_p	Closed loop subspace matrix in equation (3.9).	3.2
L_u	Subspace matrix containing the process Markov parameters.	2.2
L_v	Subspace matrix containing the measured disturbances Markov parameters.	2.2
L_u^{CL}	Closed loop subspace matrix between $M_p \longrightarrow U_f$	3.2
L_{ur}^{CL}	Closed loop subspace matrix between $R_f \longrightarrow U_f$	3.2
L_{ue}^{CL}	Closed loop subspace matrix between $E_f \longrightarrow U_f$	3.2
L_w	Subspace matrix corresponding to the states.	2.2
L_w^b	Subspace matrix corresponding to the states for system in equations (4.36)-(4.37).	4.3.2
L_w°	Modified subspace matrix defined in equation 4.30.	4.3.1
$L_w^\pm$	Modified subspace matrix defined in equation 4.46.	4.3.2

L_y^{CL}	Closed loop subspace matrix between $M_p \longrightarrow Y_f$	3.2
L_{yr}^{CL}	Closed loop subspace matrix between $R_f \longrightarrow Y_f$	3.2
L_{ye}^{CL}	Closed loop subspace matrix between $E_f \longrightarrow Y_f$	3.2
m	Number of system output(s).	2.2
M_p	$\begin{bmatrix} Y_p \\ U_p \\ R_p \end{bmatrix}$	3.2
n	Number of system state(s).	2.2
N	Number of block-rows taken in the block-Hankel matrices.	2.2
Q	Feedback controller on system in equations (2.1)-(2.2).	3.2
Q_k	Output weighting matrix	9.5.5
r_k	System measured disturbance(s) at sampling instant k.	3.2
R_f	*future* Data Hankel matrix for r_k.	3.2
R_k	Input weighting matrix	9.5.6
R_p	*past* Data Hankel matrix for r_k	3.2
S	Covariance matrix of the system unmeasured disturbances, $E[e_k e_k^T]$.	2.2
S_A	Column vector obtained from the covariance matrix Σ_A.	6.4
S_{N_2,N_u}	Dynamic matrix with (N_2-block rows and N_u-block columns) of step-response coefficients.	4.3.1
u_k	System input(s) at sampling instant k.	2.2
U, S, V	Matrices of a single value decomposition	
u_{lqg}	Process input variance if an LQG-controller were implemented on the process.	5.3
U_p	*past* Data Hankel matrix for u_k	2.2
U_f	*future* Data Hankel matrix for u_k	2.2
v_k	System measured disturbance(s) at sampling instant k.	4.3.2
V_p	*past* Data Hankel matrix for v_k	4.3.2
V_f	*future* Data Hankel matrix for v_k	4.3.2

W_p	$\begin{bmatrix} Y_p \\ U_p \end{bmatrix}$	2.2
W_p^b	$\begin{bmatrix} Y_p \\ U_p \\ V_p \end{bmatrix}$	4.3.2
x_k	System states(s) at sampling instant k.	2.2
x_k^c	Controller state(s) at sampling instant k.	3.2
X_p	*past* state sequence of the system in equations (2.1)-(2.2), $\begin{bmatrix} x_0 & x_1 & \dots & x_{j-1} \end{bmatrix}$	2.2
X_p^c	*past* state sequence of the system in equations (3.2)-(3.3), $\begin{bmatrix} x_0^c & x_1^c & \dots & {}^c x_{j-1} \end{bmatrix}$	3.2
X_f	*future* state sequence of the system in equations (2.1)-(2.2), $\begin{bmatrix} x_N & x_{N+1} & \dots & x_{N+j-1} \end{bmatrix}$	2.2
X_f^c	*future* state sequence of the system in equations (3.2)-(3.3), $\begin{bmatrix} x_N^c & x_{N+1}^c & \dots & x_{N+j-1}^c \end{bmatrix}$	3.2
y_{lqg}	Process output variance if an LQG-controller were implemented on the process.	5.3
y_k	System output(s) at sampling instant k.	2.2
Y_p	*past* Data Hankel matrix for y_k	2.2
Y_f	*future* Data Hankel matrix for y_k	2.2

Greek letters

$(\eta)^{fb}$, $(\eta)^{ff\&fb}$	LQG-benchmark based controller performance indices with respect to the process output variance.	5.5
Γ_N	Extended observability matrix of the system in equations (2.1)-(2.2).	2.2
Γ_N^b	Extended observability matrix of the system in equations (5.33)-(5.34).	2.2
Γ_N^c	Extended observability matrix of the controller in equations (3.2)-(3.3).	3.2
Δ	Differencing operator $(1 - z^{-1})$.	4.2
Δ_N	Reversed extended controllability matrix of $\{A, B\}$.	2.2
Δ_N^c	Reversed extended controllability matrix of $\{A_c, B_c\}$.	3.2
Δ_N^s	Reversed extended controllability matrix of $\{A, K\}$.	2.2
Υ	Matrix used in the noise model tuning.	4.4

Chapter 1

Introduction

1.1 An overview of objectives of this thesis

A typical industrial plant can contain thousands of controllers ranging from proportional, integral and differentiator (PI/PID) controllers to the more advanced model predictive controllers (MPC) such as dynamic matrix control (DMC) [15, 16], quadratic dynamic matrix controller (QDMC) [14, 30], robust multivariate predictive controller (RMPCT) [78], generalized predictive controller (GPC) [12, 13], etc. With a goal towards optimal performance, energy conservation and cost effectiveness of the process operations in the industry, design of optimal controllers and controller performance assessment have received much attention in both the industry and the academia. Typically a 'model' or some sort of mathematical representation of the process *and* the controller objective are required not only for designing suitable controllers but also for analyzing the controller performance. For predictive controllers, which use a model of the system to make predictions, model identification forms the critical part of the controller design. Identification aims at finding a mathematical model from the measurement record of inputs and outputs of a system [87]. Parametric model, such as a transfer function or a state space model, identification involves obtaining reduced order models of a pre-specified structure for a system which could be of a very high order and complexity. Non-parametric

modelling approaches, such as impulse/ step response coefficient forms and frequency domain based methods, can also be found in the literature for controller design and performance analysis. More importantly non-parametric model based controller design and analysis tools have been successfully used in the industry. Identification of parametric models for the process is typically used as a first step in the controller design although the transfer function may have to be eventually converted into a non-parametric model form for obtaining the control law. Data driven approaches to obtaining the prediction matrices used in the controller design directly from the process data, and avoiding the intermediate parametric model identification step, is an area of active research in recent years.

Conventionally prediction error methods are used to identify parametric models. Subspace identification methods, with their computational advantages, have emerged as a powerful alternative to the prediction error methods over the last decade. Subspace identification methods estimate state-space representation directly from the input-output data and eliminate certain constraints of the prediction error methods such as priori structure selection and non-linear optimization. Subspace identification methods identify certain matrices, which capture the correlations between the process inputs and outputs in non-parametric coefficients form, as a first step by regressing the process input-output data. Lower order state space system matrices of a pre-specified structure are then obtained from these intermediate matrices. Since these intermediate matrices can be used to derive a predictor for the system, there have been recent attempts to design predictive controllers directly from these matrices. This thesis attempts to address some of the issues previously ignored in the subspace matrices based design of predictive controllers and expands the data driven subspace approach to other important process control areas such as closed loop identification and controller performance assessment.

As much as model identification is a critical pre-step to design optimal predictive controllers, performance analysis is an important post-step to ensure optimal performance of the designed controller in operation. With the process modifications

over time the models used in the controller may need to be revisited and re-identified to improve the controller performance. Open loop model identification may not be possible whenever model changes are detected. Closed loop identification is a more feasible solution under such a scenario. Hence, closed loop identification, predictive controller design and controller performance assessment are closely related areas for industrial control applications.

The focus of the research presented in this thesis is the data driven subspace approach. The subspace matrices based approach is used for the closed loop estimation of the dynamic matrix and noise model; predictive controller design without identifying parametric models; controller performance analysis using the Linear Quadratic Gaussian (LQG) controller as a benchmark; estimation of multivariate Minimum Variance Control (MVC) benchmark without calculating the interactor matrix; and feedforward controller performance analysis for identification of the important disturbance variables. The approach used in this thesis can be considered a 'data driven approach' in the sense that no traditional parametric models are used. Certain intermediate subspace matrices obtained directly from the process data, which are identified as a first step in the subspace identification methods, are used in the closed loop identification, controller design and performance analysis.

If some of the disturbance variables are measurable, analysis of feedforward control performance is a worthwhile study. The subspace matrix corresponding to the measured disturbance variables can be easily estimated under closed loop conditions, if the measured disturbances are assumed to be uncorrelated with setpoint. These parametric matrices can be used to incorporate feedforward control into the optimal LQG controller and also in the minimum variance controller. This provides a means for the profit analysis on implementation of feedforward control on the process.

One of the important tasks that arise during the design and implementation of feedforward control is the selection of important disturbance variables from the many available measurements. Dynamic multivariate analysis of variance of the

measured disturbance variables using principal component analysis provides a tool for finding those variables which contribute most to the process output variance.

Although many predictive control strategies have been proposed in the literature, for practical reasons, industries rely on the commercial advanced control software packages such as Dynamic Matrix Control (DMC) (Aspen Tech) and Robust Multi-variable Predictive Control Technology (RMPCT) (Honeywell), etc. for the implementation of advanced control systems. For the performance analysis of such advanced control applications understanding the working mechanism of these control systems is necessary. Variances based controller performance indices may be used as a measure of the controller performance. Comparing the designed controller objective with the achieved controller performance has also been suggested for the performance assessment of industrial model predictive controllers.

In this thesis a study is presented on the implementation of an MPC application on an industrial process and the performance analysis of the advanced control application with respect to the objective of the implementation of the controller. The significance of the application is that the performance of the controller can be measured in terms of the process operation in the optimal operating range, rather than in terms of the variances of the process variables.

1.2 Contributions of this thesis

This thesis combines both theoretical work and industrial applications in the three important areas of advanced control applications; identification, design and performance analysis, using a new data-driven subspace approach.

1.2.1 Contributions to the theory

The main theoretical contributions include:

1. Derivation of subspace matrices based identification algorithm for the

identification of dynamic matrix and noise model from closed loop data.

2. Extension of the above closed loop identification algorithm to the case of measured disturbances in the system.

3. Derivation of a design methodology for obtaining a GPC-type predictive controller directly from the process input-output data using the subspace matrices.

4. Inclusion of all the 'bells and rings', such as,

   a. Integrator in the controller law

   b. Feedforward control

   c. Noise model tuning

required for the practical implementation of the subspace matrices based predictive controller.

5. Proof of equivalence between the subspace matrices based predictive controller and the generalized Predictive Controller (GPC).

6. Derivation of expressions for calculation of the LQG-benchmark of process input and output from the subspace matrices estimated using the closed loop data.

7. Extension of the LQG-benchmark to the case of feedforward plus feedback controller on the system.

8. Deriving the LQG-benchmark based performance indicators for controller performance analysis.

9. Derivation of expressions for the calculation of confidence intervals for the LQG-benchmark obtained using the subspace matrices approach.

10. Derivation of a methodology for obtaining the multivariate MVC-benchmark which does not require the calculation of the interactor matrix.

11. Proof of equivalence of subspace matrices based approach to the transfer function approach in deriving the multivariate MVC-benchmark for controller performance

assessment.

12. Extension of the subspace matrices based MVC-benchmark calculation to that of feedforward control.

13. Dynamic analysis of variance for the selection of variables for feedforward control, both from the transfer function approach and the subspace matrices approach.

14. Derivation of MPC-relevant benchmark variance expressions for performance assessment of advanced model predictive controllers.

1.2.2 Contributions via practical applications

The main practical applications include:

1. Application of the subspace matrices based closed loop identification method on a pilot scale process.

2. Application of the subspace matrices based predictive controller on a multivariate pilot scale process.

3. Application of the LQG-benchmark based controller performance analysis on a pilot scale process.

4. An industrial case study with a multi-faceted analysis of the performance of an advanced controller application on a settling process. Application of the MVC-benchmark and LQG-benchmark based performance analysis techniques on the industrial application is also shown.

1.3 Organization of the thesis

This thesis is organized as follows. Chapter 2 gives an overview of the existing subspace identification methods for the open-loop and closed-loop data. This

chapter provides the basic background required for understanding the rest of the thesis. The rest of the thesis is organized in the sequence of identification, controller design and performance analysis techniques.

In section 3, the identification method for obtaining the process dynamic matrix and the noise model from closed-loop data with setpoint excitation is described. The main results of the paper are demonstrated through an application on a pilot scale process. In section 4, the design of a predictive controller directly from the subspace matrices is described with a demonstration on a multivariate pilot scale process.

Section 5 describes the methodology for LQG-benchmark based controller performance analysis using the subspace matrices and the routine process operating data. In section 6 the expressions for obtaining the confidence intervals for the LQG-benchmark curve are derived.

In section 7, the methodology for obtaining the MVC-benchmark for process output without the calculation of the interactor matrix is described. It is also theoretically shown in the same chapter that the derived subspace matrices based methodology is equivalent to the interactor matrix based transfer function approach derived in [42]. In section 8 the analytical procedure for the dynamic multivariate analysis of the process output variance from the feed-forward variables is described. This analysis is based on the numerically robust singular value decomposition (SVD) methodology.

In section 9, an industrial case study of the controller performance analysis on a settling process is presented.

This thesis has been written in a paper-format in accordance with the rules and regulations of the Faculty of Graduate Studies and Research, University of Alberta. Many of the chapters have appeared or are to appear in archival journals and conference proceedings. In order to link the different chapters, there is some overlap and redundancy of material. This has been done to ensure completeness and cohesiveness of the thesis material and help the reader understand the material easily.

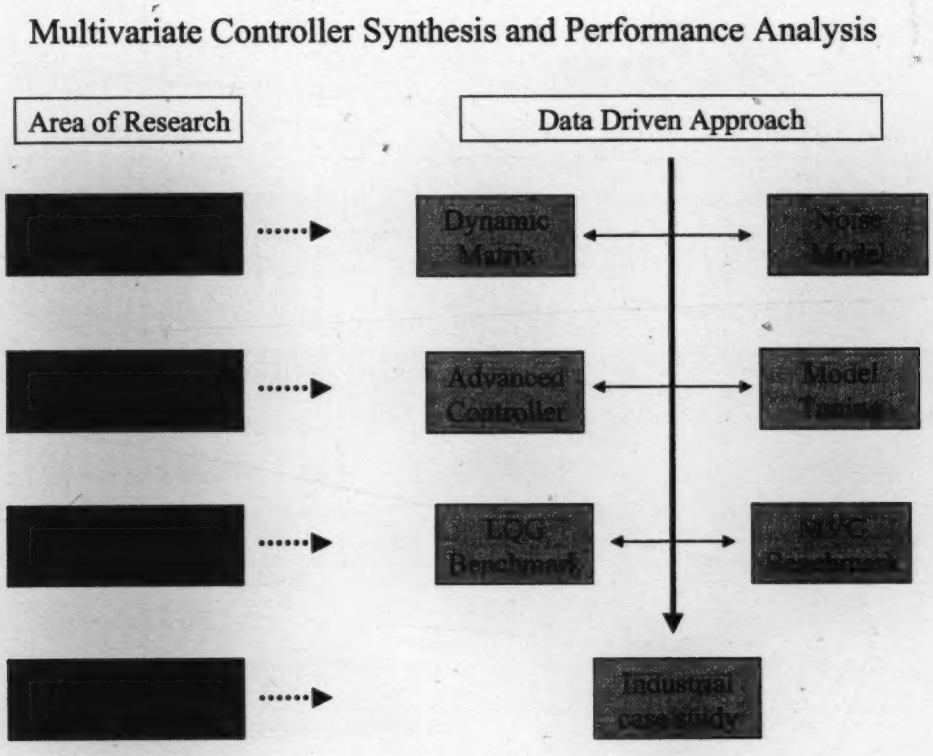


Figure 1.1: Research Overview

Chapter 2

Review on subspace identification methods

2.1 Introduction

Conventionally, a system is modelled by a transfer function, which is a fractional representation of two polynomials with real coefficients, identified using an iterative optimization scheme for a nonlinear least-squares fit to the data [80]. Subspace identification methods offer an alternative for the identification of a 'model' for the systems and are based on computational tools such as QR-factorization and SVD, which make them intrinsically robust from a numerical point of view. Subspace identification methods are also non-iterative procedures (avoiding local minima and convergence problems) and can be easily converted into an adaptive version of model identification [87]. Subspace identification methods are intrinsically suitable for multivariate systems identification compared to the prediction error methods. This section gives a brief description of the main types of subspace identification methods available in the literature. A more detailed presentation of these methods can be found in the standard book on subspace identification such as [93] and the special issues of the journals *Automatica* [1, 2] and *Signal Processing* [3], along with the references therein. Other variations in subspace identification

techniques, such as DSR [102, 105], errors-in-variables (EIV) approach, and the use of weighting matrices in subspace identification are not presented here and can be found elsewhere.

Although the principal goal of the methods described in this chapter is to identify the state space system matrices { A, B, C, D }, certain subspace matrices are first calculated as an intermediate step. The rest of this thesis uses *only* these intermediate matrices for identification, control and analysis purposes. Nevertheless, this chapter provides some insights into the special features of the subspace identification methods.

2.2 Subspace matrices description

A linear time-invariant system can be represented in a state space innovations form as:

$$x_{k+1} = Ax_k + Bu_k + Ke_k \tag{2.1}$$

$$y_k = Cx_k + Du_k + e_k \tag{2.2}$$

where $u_k(l \times 1)$, $y_k(m \times 1)$ and $x_k(n \times 1)$ are the vectors of l-inputs, m-outputs and n-states respectively of the system. K is the Kalman filter gain and $e_k(m \times 1)$ is an unknown innovations process of white noise with the following covariance matrix:

$$E[e_k e_k^T] = S \tag{2.3}$$

e_k is assumed to be stationary. The state space system matrices A, B, C, D, K and S are of the dimension $(n \times n)$, $(n \times l)$, $(m \times n)$, $(m \times l)$, $(n \times m)$ and $(m \times m)$ respectively. Subspace identification starts with obtaining the measurements of the inputs and the outputs u_k, y_k for $k \in \{0, 1, ..., 2N + j - 2)\}$. The data is then arranged in the block Hankel matrices form of N- block rows and j- block columns,

defined as:

$$U_p = \begin{bmatrix} u_0 & u_1 & \cdots & u_{j-1} \\ u_1 & u_2 & \cdots & u_j \\ \cdots & \cdots & \cdots & \cdots \\ u_{N-1} & u_N & \cdots & u_{N+j-2} \end{bmatrix} \tag{2.4}$$

$$U_f = \begin{bmatrix} u_N & u_{N+1} & \cdots & u_{N+j-1} \\ u_{N+1} & u_{N+2} & \cdots & u_{N+j} \\ \cdots & \cdots & \cdots & \cdots \\ u_{2N-1} & u_{2N} & \cdots & u_{2N+j-2} \end{bmatrix} \tag{2.5}$$

$$Y_p = \begin{bmatrix} y_0 & y_1 & \cdots & y_{j-1} \\ y_1 & y_2 & \cdots & y_j \\ \cdots & \cdots & \cdots & \cdots \\ y_{N-1} & y_N & \cdots & y_{N+j-2} \end{bmatrix} \tag{2.6}$$

$$Y_f = \begin{bmatrix} y_N & y_{N+1} & \cdots & y_{N+j-1} \\ y_{N+1} & y_{N+2} & \cdots & y_{N+j} \\ \cdots & \cdots & \cdots & \cdots \\ y_{2N-1} & y_{2N} & \cdots & y_{2N+j-2} \end{bmatrix} \tag{2.7}$$

N and j should be chosen 'sufficiently large' (so that the data Hankel matrices contain enough information on the system), and typically $j \gg \max(mN, lN)$ ('very rectangular' block Hankel matrices), as this reduces both the computational load and noise sensitivity [87]. Each element in the above data Hankel matrices is a column vector of inputs and outputs, i.e., $u_i = \begin{bmatrix} u_{i1} \\ \cdots \\ u_{il} \end{bmatrix}$ and $y_i = \begin{bmatrix} y_{i1} \\ \cdots \\ y_{im} \end{bmatrix}$. The subscripts p and f denote 'past' and 'future' in the sense that the corresponding elements in the p-matrices and f-matrices are separated by N sample instants. The past and future state sequences are similarly defined as

$$X_p = \begin{bmatrix} x_0 & x_1 & \cdots & x_{j-1} \end{bmatrix} \tag{2.8}$$

$$X_f = \begin{bmatrix} x_N & x_{N+1} & \cdots & x_{N+j-1} \end{bmatrix} \tag{2.9}$$

The following matrix input-output equations can be formed [87] by recursively substituting the system equations (2.1)-(2.2)

$$X_f = A^N X_p + \Delta_N U_p + \Delta_N^s E_p \tag{2.10}$$

$$Y_p = \Gamma_N X_p + H_N U_p + H_N^s E_p \tag{2.11}$$

$$Y_f = \Gamma_N X_f + H_N U_f + H_N^s E_f \tag{2.12}$$

where Γ_N is the extended observability matrix, H_N and H_N^s are the lower triangular Toeplitz matrices containing the impulse response coefficients/Markov parameters of the system due to the deterministic inputs and the unknown stochastic inputs respectively. Δ_N and Δ_N^s are the reversed extended controllability matrices of $\{A, B\}$ and $\{A, K\}$ respectively. The subscript, N, follows from the number of block-rows taken in the block Hankel matrices as shown above. The system related matrices are expressed as

$$\Gamma_N = \begin{bmatrix} C \\ CA \\ \dots \\ CA^{N-1} \end{bmatrix} \tag{2.13}$$

$$H_N = \begin{bmatrix} D & 0 & \dots & 0 \\ CB & D & \dots & 0 \\ \dots & \dots & \dots & \dots \\ CA^{N-2}B & CA^{N-3}B & \dots & D \end{bmatrix} \tag{2.14}$$

$$H_N^s = \begin{bmatrix} I_m & 0 & \dots & 0 \\ CK & I_m & \dots & 0 \\ \dots & \dots & \dots & \dots \\ CA^{N-2}K & CA^{N-3}K & \dots & I_m \end{bmatrix} \tag{2.15}$$

$$\Delta_N = \begin{bmatrix} A^{N-1}B & A^{N-2}B & \dots & B \end{bmatrix} \tag{2.16}$$

$$\Delta_N^s = \begin{bmatrix} A^{N-1}K & A^{N-2}K & \dots & K \end{bmatrix} \tag{2.17}$$

By substituting equation (2.11) in equation (2.10) we can write

$$X_f = \left[A^N \Gamma_N^\dagger \quad \left(\Delta_N - A^N \Gamma_N^\dagger H_N\right) \quad \left(\Delta_N^s - A^N \Gamma_N^\dagger H_N^s\right) \right] \begin{bmatrix} Y_p \\ U_p \\ E_p \end{bmatrix} \tag{2.18}$$

where † represents the Moore-Penrose pseudo-inverse. Substituting equation (2.18) in equation (2.12), we can write

$$Y_f = L_w W_p + L_u U_f + L_e E_f \tag{2.19}$$

where

$L_w \longrightarrow$ *subspace matrix corresponding to the states*

$L_u \longrightarrow$ *subspace matrix corresponding to the deterministic inputs*

$L_e \longrightarrow$ *subspace matrix corresponding to the stochastic inputs*

The matrix W_p is formed by the concatenation of Y_p and U_p as

$$W_p = \begin{bmatrix} Y_p \\ U_p \end{bmatrix} \tag{2.20}$$

If we take a closer look by expanding subspace matrices (see figure 2.1), we observe that the first row of the subspace matrix L_w is the ARX model of the system. As $N \longrightarrow \infty$, the last row of subspace matrices L_u and L_e transform into impulse response models for the process and disturbance respectively.

2.3 Regression techniques

The subspace matrices can be identified from the data Hankel matrices using regression techniques. Several regression techniques, including the usage of

weighting matrices, have been suggested in the literature for numerical advantages and special cases of data collection. The simplest regression method is the least squares method. The conditions on the process data for using the regressions techniques, as evident from equation (2.19), are as follows:

i. The deterministic input u_k is uncorrelated with e_k, and e_k is not identically zero,

ii. u_k is persistently exciting of order $2N$, and

iii. The number of measurements goes to infinity, $j \longrightarrow \infty$.

The open loop identification of the subspace matrices using the least squares solution involves finding the prediction of the future outputs Y_f using a linear predictor:

$$\hat{Y}_f = L_w W_p + L_u U_f$$

The least squares prediction $\hat{Y}_f$ can be found by solving an optimization problem:

$$\min_{L_w, L_u} \| Y_f - \begin{pmatrix} L_w & L_u \end{pmatrix} \begin{pmatrix} W_p \\ U_f \end{pmatrix} \|_F^2$$

$\hat{Y}_f$ is found by the orthogonal projection of the row space of Y_f into the row space spanned by W_p and U_f defined as [93]:

$$\hat{Y}_f = Y_f / \begin{bmatrix} W_p \\ U_f \end{bmatrix} \tag{2.21}$$

$$\begin{bmatrix} L_w & L_u \end{bmatrix} = Y_f \begin{bmatrix} W_p \\ U_f \end{bmatrix}^{\dagger} = Y_f \begin{bmatrix} W_p^T & U_f^T \end{bmatrix} \left(\begin{bmatrix} W_p \\ U_f \end{bmatrix} \begin{bmatrix} W_p & U_f \end{bmatrix} \right)^{-1} \tag{2.22}$$

This projection can also be implemented in a numerically robust way with a QR-decomposition [91, 92, 93, 102, 119] or using PLS [105].

2.4 Open loop subspace identification methods

In principle, the subspace methods can be described as estimation of a rational covariance model from observed data followed by stochastic realization [79]. In the

open loop subspace state space identification methods, the sequence of the future states, X_f, and the extended observability matrix, Γ_N, are estimated using equations (2.11)-(2.12) and the estimation requires that the pair $\{A, C\}$ is completely observable since only the modes that are observable can be identified from observed I/O-data. Furthermore, the pair $\{A, \begin{bmatrix} B & K \end{bmatrix}\}$ requires to be controllable. This implies that all modes are sufficiently excited (persistent excitation). Note that even though the deterministic subsystem can have unstable modes, the excitation u_k has to be chosen in such a way that the deterministic states and output are bounded for all times. Also the deterministic and stochastic subsystem may have common or completely decoupled input-output dynamics. If the pair $\{A, C\}$ are observable, then the rank of Γ_N is equal to the state order n.

Subspace identification involves estimating a basis for the states of the system from the data Hankel matrices. It must be remembered that the states identified using these techniques do not have any physical meaning. The different subspace identification techniques available in the literature differ in the manner in which this basis of the state space is estimated. The different choices for a basis differ in a transformational matrix T that transforms a model $\{A, B, C, D\}$ into an equivalent model $\{AT, T^{-1}B, CT, D\}$ [61].

The numerical tools used in the estimation of this basis range from SVD (used in [87], N4SID [93]), QR-decomposition (used in MOESP [119, 120, 121]), canonical variables (used in CVA [72, 73, 74, 75, 76, 77, 107]), etc. Some subspace identification methods also differ on how the disturbances are characterized.

2.4.1 N4SID

N4SID can be explained as a linear regression multi-step ahead prediction error method with certain rank constraints [52]. Using the SVD approach the reduced order observability matrix, Γ_n, and the non-steady state Kalman filter estimate of state sequence, $\hat{X}_f$, are obtained. Under the assumptions listed in section 2.3 the

following oblique projection is defined:

$$O_N = Y_f/_{U_f} W_P \tag{2.23}$$

$$= \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} S_n & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} \tag{2.24}$$

where the last '$(N-n)$' singular values are approximately zero (or very small compared to the first 'n' singular values) and the projection $\mathcal{A}/_{\mathcal{B}}\mathcal{C}$ is defined as the oblique projection of the row space of $\mathcal{A} \in \Re^{p \times j}$ along the row space of $\mathcal{B} \in \Re^{q \times j}$ on the row space of $\mathcal{C} \in \Re^{r \times j}$:

$$\mathcal{A}/_{\mathcal{B}}\mathcal{C} = A \begin{bmatrix} \mathcal{C}^T & \mathcal{B}^T \end{bmatrix} \left\{ \begin{bmatrix} \mathcal{C}\mathcal{C}^T & \mathcal{C}\mathcal{B}^T \\ \mathcal{B}\mathcal{C}^T & \mathcal{B}\mathcal{B}^T \end{bmatrix}^{\dagger} \right\} \tag{2.25}$$

The matrix O_N is equal to the product of the extended observability matrix and the estimated Kalman filter state sequence:

$$O_N = \Gamma_N \hat{X}_N \tag{2.26}$$

Therefore Γ_n and $\hat{X}_n$ are determined using the SVD as

$$\Gamma_n = U_1 S_n^{\frac{1}{2}} \tag{2.27}$$

$$\hat{X}_n = S_n^{\frac{1}{2}} V_1^T \tag{2.28}$$

See [93] for the derivation of the above result. As can be seen, the system order is obtained using the SVD function.

The state space matrices A, B, C, D, K and S can then be estimated by using either the reduced order observability matrix or the state sequence estimate [11, 73, 75, 91, 92, 93, 102, 105].

2.4.2 MOESP

MOESP [119] uses QR-decomposition, followed by SVD, as the numerical tools to obtain the extended observability matrix:

$$\begin{bmatrix} U_f \\ W_p \\ Y_f \end{bmatrix} = \begin{bmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & L_{33} \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \end{bmatrix} \tag{2.29}$$

and

$$L_{32} = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} S_n & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} \tag{2.30}$$

The reduced order observability matrix of the system is derived as

$$\Gamma_n = U_1 S_n^{\frac{1}{2}} \tag{2.31}$$

$$\tag{2.32}$$

Hence the system order is obtained using the SVD. From the reduced order observability matrix, Γ_n, the system matrices A and C are derived. Using $\{A, C\}$ and Γ_n with the process input-output data, the rest of the system matrices are identified.

2.4.3 CVA

In this method canonical variables are used to provide an ordered basis of the state space, ordered in terms of the predictive ability of the states [72, 73, 74, 75, 76, 77, 107]. The canonical correlations between 'the past, W_p, conditional to the future inputs, U_f' and 'the future outputs, Y_f, conditional to the future inputs, U_f' are used as the basis of the state space. If k is the true and finite state order and there is sufficient information to reliably determine the order k, then the first k canonical variables give an optimal selection of system states in terms of maximizing the likelihood function [73]. The CVA [75, 73, 107] algorithm uses Akaike Information Criteria (AIC) to determine the state order.

Note that Van Overschee and De Moor [92] have classified the above three methods as special cases under a unifying framework and the difference between the three algorithms are expressed in terms of the weighting matrices involved in the SVD. Other variations in the subspace algorithms will also calculate the same result (upto within a similarity transformation) provided the same system order n is chosen and the number of data points go to infinity (since all algorithms are asymptotically unbiased) [92].

2.5 Closed loop subspace identification methods

Closed loop data cannot be used with the above methods for model identification because of the correlations between U_f and E_f in equation (2.19) (see figure 2.2). Several approaches have been used to overcome this constraint. The various closed loop subspace identification approaches reported in literature can be summarized as follows:

2.5.1 N4SID approach

The N4SID approach [94] makes use of only the process input-output data ($C_1 = 0$; $C_2 = C$; $w(t) = 0$; $r(t)$ does not have to be measured) and the knowledge of the first few impulse response (Markov parameters for multivariate systems) of the controller. The controller equations can be written as

$$x^c_{k+1} = A^c x^c_k + B^c y_k \tag{2.33}$$

$$u_k = r_k - C^c x^c_k - D^c y_k \tag{2.34}$$

where $\{ A^c, B^c, C^c, D^c \}$ are the controller system matrices. The controller cannot be unstable.

Following the notation in [94], define $Z_{0|i-1}$ as the data Hankel matrix of the variable z_k with i-block rows and j-block columns ($Z_{0|i-1}$ is built similar to the data Hankel matrices defined in equations (2.4)-(2.7)). Certain pseudo data matrices are

constructed from the input and output data Hankel matrices using the knowledge of the controller IR/Markov parameters:

$$N_{p|q} = U_{0|2N-1} + H^c_{2N} Y_{0|2N-1} \tag{2.35}$$

$$M_{p|q} = U_{p|q} + H^c_{p-q+1} Y_{p|q} \tag{2.36}$$

where $0 \leq p \leq q \leq 2N-1$, H^c is the lower triangular block-Toeplitz matrix for the controller, and $N_{p|q}$ and $M_{p|q}$ (not block Hankel matrices) are uncorrelated with the disturbances. Under the assumptions that:

1. r_k is uncorrelated with the disturbances.
2. The matrix $N_{0|2N-1}$ has a full row rank $2mN$.
3. $j \longrightarrow \infty$ and
4. The closed loop problem is well posed, i.e. $(I_l + DD_c)$ is invertible, then

$$Y_{N|2N-1}/\begin{bmatrix} U_{0|N-1} \\ Y_{0|N-1} \\ M_{N|2N-1} \end{bmatrix} = Y_{N|2N-1}/\begin{bmatrix} N_{0|2N-1} \\ Y_{N|N-1} \end{bmatrix} = P_N \hat{X}_N + Q_N M_{N|2N-1} \tag{2.37}$$

Therefore, define the oblique projection

$$O_N = Y_{N|2N-1}/_{M_{N|2N-1}} \begin{bmatrix} U_{0|N-1} \\ Y_{0|N-1} \end{bmatrix} \tag{2.38}$$

$$= \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} S_n & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} \tag{2.39}$$

where state order = rank(O_N) and estimated states sequence $\hat{X}_N = S_n^{\frac{1}{2}} V_1^T$. A state space model for the process is subsequently identified from the estimated states.

The limitation of this method is the requirement of the knowledge of the controller. In industrial systems, the accurate knowledge of the IR coefficients/ Markov parameters of the controller may not always be available.

2.5.2 MOESP/ CVA approaches

The overall strategy of these methods is similar to the joint input-output identification strategy. These methods do not require the knowledge of the controller. Apart from setpoint excitation, these approaches [73, 113, 122, 71] need an additional dither signal excitation, $w(k)$, added to the controller output, $u_e(k)$, to make the process input independent of noise. $C_1 = C$ and $C_2 = 0$. $r(k)$ is the setpoint which is a white noise sequence. $v(k)$ is the white noise (disturbance) added to the process output, y_k, through a noise model, F. Then the measurable input vector is $\begin{bmatrix} w(k) & r(k) \end{bmatrix}$ and measurable output vector is $\begin{bmatrix} u_e(k) & e(k) & u(k) & y(k) \end{bmatrix}$, where $u_e = Ce = u - w$ and $e = r - y$. Using this information a global state space model is first identified using MOESP technique. The global state space model is denoted as:

$$x(k+1) = Ax(k) + \begin{bmatrix} B_1 & B_2 \end{bmatrix} \begin{bmatrix} w(k) \\ r(k) \end{bmatrix} \tag{2.40}$$

$$\begin{bmatrix} u_e(k) \\ e(k) \\ u(k) \\ y(k) \end{bmatrix} = \begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \end{bmatrix} x(k) + \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \\ D_{31} & D_{32} \\ D_{41} & D_{42} \end{bmatrix} \begin{bmatrix} w(k) \\ r(k) \end{bmatrix} \tag{2.41}$$

$x(k)$ has an order $n_p + n_c$, where n_p is the order of the process, P, and n_c is the order of the controller, C. The state space representations between 'w and y' and 'w and u' are given by the system matrices $[A, B_1, C_4, D_{41}]$ and $[A, B_1, C_3, D_{31}]$ respectively [122].

Using the rules for concatenating and inverting of state space models, the state space models for the process, P, and the controller, C, are obtained as:

$$\begin{aligned} P &= [A, B_1, C_4, D_{41}][A, B_1, C_3, D_{31}]^{-1} \\ &= [A, B_1, C_4, D_{41}][A - B_1 D_{31}^{-1} C_3, B_1 D_{31}^{-1}, -D_{31}^{-1} C_3, D_{31}^{-1}] \\ &= \left(\begin{bmatrix} A & -B_1 D_{31}^{-1} C_3 \\ 0 & A - B_1 D_{31}^{-1} D_3 \end{bmatrix}, \begin{bmatrix} B_1 D_{31}^{-1} \\ B_1 D_{31}^{-1} \end{bmatrix}, \begin{bmatrix} C_4 & -D_{41} D_{31}^{-1} C_3 \end{bmatrix}, D_{41} D_{31}^{-1} \right) \end{aligned} \tag{2.42}$$

and

$$
\begin{aligned}
C &= [A, B_2, C_3, D_{32}][A, B_2, C_2, D_{22}]^{-1} \\
&= \left(\begin{bmatrix} A & -B_2 D_{22}^{-1} C_2 \\ 0 & A - B_2 D_{22}^{-1} D_2 \end{bmatrix}, \begin{bmatrix} B_2 D_{22}^{-1} \\ B_2 D_{22}^{-1} \end{bmatrix}, \begin{bmatrix} C_3 & -D_{32} D_{22}^{-1} C_2 \end{bmatrix}, D_{32} D_{22}^{-1} \right)
\end{aligned} \tag{2.43}
$$

The overall deterministic state space model is first identified using MOESP. The order selection in this step needs the specification of the sum of process model order and controller order [122]. The indiyidual plant and controller model orders are determined/selected in the subsequent model reduction step.

CVA approach [71] uses practically the same approach as MOESP except that the CVA algorithm is used to identify the state space matrices of the global closed loop system, with AIC used in the selection of the state order. The state space representation is then converted to the transfer function representation followed by the concatenation and inversion of the closed loop transfer functions to obtain the open-loop process transfer function.

2.5.3 Ljung and McKelvey's approach via estimated predictors

In the subspace algorithms, the state space matrices are identified from the estimated states. Hence this approach tries to reconstruct the 'Kalman states' from the past inputs and outputs by picking a basis from the estimated predictors. As a first step [81] an (na, nb)-order ARX-model is identified from the closed loop input and output data. The ARX model is used to calculate the j-step ahead predictors $\hat{y}(t+j|t)$ from data by replacing 'future' $u(t)$ in this prediction with zero. The vector of j-step ahead predictors is formed as

$$
Y_m(t) = \begin{bmatrix} \hat{y}(t+1|t) \\ \ldots \\ \hat{y}(t+m|t) \end{bmatrix} \tag{2.44}
$$

The number of states, $n = rank[Y_m(t)]$ for $m \geq n$. The states are obtained using the projection

$$x(t) \;=\; LY_m(t) \tag{2.45}$$

for some matrix L. In summary, the identification strategy is,

1. Find $\hat{y}(t+j|t)$ for $j = 1, ..., m$.
2. Form $Y_m(t)$ for $t = 1, ..., N$ and estimate its rank n.
3. Select L to obtain a well conditioned basis and $x(t) = LY_m(t)$, $t = 1, ..., N$.
4. Find the matrices $\{A, B, C, D, K\}$ by using the estimated states.

Note that the states are identified from the estimated predictors and not directly from the data, unlike the previously illustrated approaches. The authors [81] state that this method is only a 'feasible' method rather than the 'best way' of identifying systems operating in closed loop.

This concludes the review of some of the subspace identification techniques existing in the literature, which provide the background for understanding the rest of this thesis.

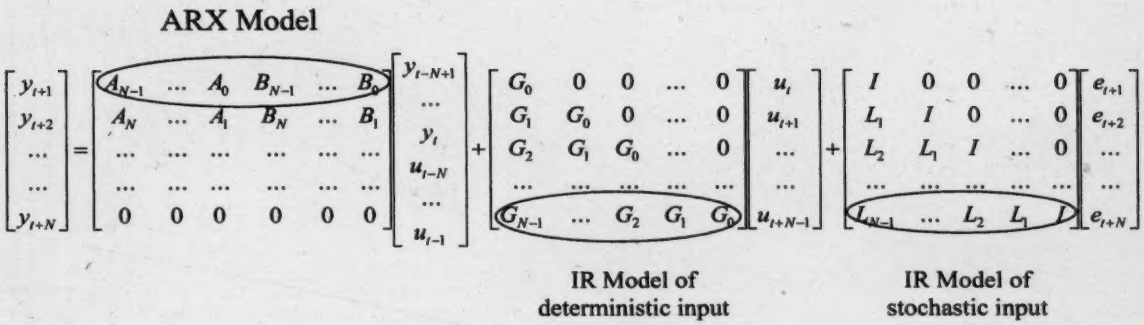


Figure 2.1: Subspace representation

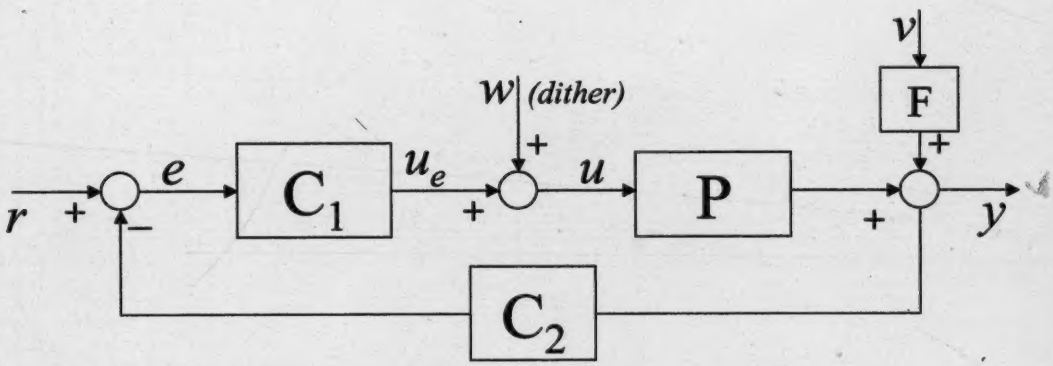


Figure 2.2: Closed loop system for subspace identification.

Chapter 3

Estimation of Dynamic matrix and noise model for MPC using closed loop data

[1]

3.1 Introduction

Model predictive controllers (MPC) have found many successful applications in process industries for more than two decades. One of the key aspects of MPC is the prediction of the future process response and minimization of the output deviation from the setpoint by manipulating the inputs. A model for the process is required to make these predictions based on the past data. Hence an MPC design starts with first identifying a nominal model for the process. One of the industrially successful predictive control schemes is the Dynamic Matrix Controller or DMC, which explicitly uses a lower triangular matrix called 'dynamic matrix' containing

[1] A version of this chapter has been published as a journal paper

R. Kadali, and B. Huang. Estimation of the Dynamic Matrix and Noise Model for Model Predictive Control Using Closed-Loop Data. *Ind. Eng. Chem. Res.*, 41(4); 842-852, 2002.

the step response coefficients corresponding to the deterministic input(s) to the process [15, 16]. Many other MPC formulations also use the dynamic matrix in one way or the other [10, 78, 88]. For constructing the dynamic matrix, in the case of DMC, a step response model for the process is first obtained from the open loop data. The step response coefficients are arranged in a specific lower triangular form known as the dynamic matrix. However, due to safety reasons and other practical limitations, open loop operation of the process may not always be possible or in some cases there may be a hidden feedback in the system. Estimation of the dynamic matrix from closed loop data is desired in such cases. It has been shown [40] that if the model is used for model-based control design then the favorable experimental conditions are actually under closed-loop condition.

Closed loop identification refers to the identification of process model and noise model using the data sampled under feedback control. Correlation between the disturbances entering the process and the input offers the fundamental limitation [4, 27, 32, 80, 81, 110] for utilizing the standard open loop identification methods with closed loop data (see figure 3.1). Several closed loop parametric model identification methods have been suggested in the literature which require either certain assumptions about the model structure or knowledge of the controller model. The closed loop identification methods found in the literature are broadly classified [27] into Direct [108], Indirect [18, 41, 118] and Joint input/output [43] identification methods. See [17, 27, 90] for a review on the features and limitations of different closed loop identification methods.

The subspace identification method is a relatively new approach used for the state space model identification. In this approach, certain subspace matrices of the process are first identified, by regression of the data Hankel matrices (refer to chapter 2), from which the state space matrices are extracted. Three subspace matrices are obtained as a first step of the subspace identification methodologies [91, 92, 93]. The subspace matrices $\{L_w,\ L_u$ and $L_e\}$ (refer to chapter 2) correspond to the states, the deterministic input(s) and the stochastic input(s) to the system. These subspace matrices are directly calculated from the input/output data matrices in

a single iteration compared to the iterative schemes used in the prediction error methods. Moreover, the subspace identification methods minimize the summation of the multi-step ahead prediction errors during the estimation of the subspace matrices. This advantage is however lost when lower order state space system matrices are estimated from the subspace matrices. Hence directly using the subspace matrices is a very appealing idea for designing the predictive controllers [23, 25, 26, 56, 59, 103, 104, 106]. The subspace matrix corresponding to the deterministic input contains the impulse response coefficients (Markov parameters for multivariate processes) of the deterministic input(s) in a lower triangular form. Similarly the subspace matrix corresponding to the stochastic input contains the impulse response coefficients/Markov parameters of the noise model. This allows the alternative approach for direct estimation of the dynamic matrix and noise model from the open loop input/output data matrices. However it has been shown [27, 80] that the open loop subspace identification methods cannot be directly applied to the closed loop data.

Identification of the subspace matrices from closed loop data has recently received an attention by a number of researchers [24, 25, 26, 81, 94]. Van Overschee and De Moor [94] proposed an N4SID (Numerical subspace state space identification) based method for closed loop subspace identification which requires the knowledge of the first N impulse response coefficients of the controller, where N is the maximum order of the state space model to be identified. The knowledge of the impulse response (IR) coefficients of the controller are required if one wants to identify all the three subspace matrices and subsequently a state space model for the system. Ljung and McKelvey [81] presented a method for the identification of subspace matrices from closed loop data using estimated predictors and stated that their algorithm is merely an illustration of a 'feasible' method rather than the 'best way' of identifying systems operating in closed loop. MOESP (MIMO output error state space model identification) and CVA (Canonical variate analysis) approaches are also proposed for the identification of a state space model using closed loop data [72, 73, 74, 75, 76, 77, 107, 113, 122]. In addition to the setpoint excitation, MOESP/

CVA approaches use an external white noise signal addition to the controller output to make it independent of the noise. A closed-loop state space model is first identified from the closed loop data from which the open loop state space matrices are retrieved. The principal goal of all the above approaches is the identification of a state space model for the system using closed loop data.

Even though the subspace identification method is used as a vehicle, the goal of the identification method from closed loop data proposed in this chapter is *not* the estimation of state space system matrices $\{A, B, C, D$ and $K\}$ but the estimation of the dynamic matrix of the process and the noise model in impulse response form. It is shown in this chapter that if we want to estimate only two of the subspace matrices, i.e. only those corresponding to the deterministic and stochastic inputs, from closed loop data, then the knowledge of the controller impulse response coefficients can be avoided. We can then obtain the process dynamic matrix from the deterministic input subspace matrix and the noise model in impulse response form from the stochastic input subspace matrix.

The method proposed in this chapter can be considered as a non-parametric approach for closed loop identification. Non-parametric model identification methods, although known to give less bias error due to less model structure and order limitations, could result in higher variance (due to the larger number of parameters) compared to parametric model identification methods. This is a tradeoff between bias error and variance error in process identification. Actual process is typically high-order and nonlinear and it is difficult to be represented by a single linear parametric model. Consequently, bias error is inevitable in practice. On the other hand, it is known that the variance error can be reduced with the increased sample size [80]. Therefore, depending on the application, for example, depending on the data sample size, one can choose to use the parametric or nonparametric identification method.

However, non-parametric model identification methods do have some practical advantages. Consider the case when we want to identify a process model for

designing and implementing a model based predictive controller. Even if we first identify a parametric model for the process (using closed or open loop process data), the process model has to be eventually transfered to a non-parametric (impulse or step response models) form for designing the MPC. The question is this additional intermediate step may introduce an additional error and inconvenience. When it comes to industrial implementation, non-parametric model based MPCs have shown a considerable success rate. The idea here is why not directly identify a non-parametric model and avoid choosing a 'model structure', which is unknown and a prerequisite for parametric model identification methods.

The remaining of the chapter is organized as follows. Section 3.2 gives the description of the estimation of the process dynamic matrix and noise model from closed loop data. Remarks on the different steps of the closed loop identification along with some guidelines for the practical implementation of the algorithm are provided in section 3.3. The closed loop identification method is extended to the case of measured disturbances in section 3.5. The application of the proposed method is explained using MATLAB simulations in section 3.6 and implementation on a pilot scale process in section 3.7. Conclusions from the above work are given in section 3.8.

3.2 Estimation of the process dynamic matrix and the noise model from closed loop data

Consider the case when the system (2.1)-(2.2) is operating under closed loop with a linear time-invariant feedback-only controller Q, expressed in transfer function form as

$$u_k = Q\ (r_k - y_k) \tag{3.1}$$

where r_k is the setpoint for the process output at the sampling instant k and $(r_k - y_k)$ is the output deviation from the setpoint. Assume that the controller does not cancel any plant dynamics. The control system can be expressed in state space

representation as

$$x^c_{(k+1)} = A_c x^c_k + B_c(r_k - y_k) \tag{3.2}$$

$$u_k = C_c x^c_k + D_c(r_k - y_k) \tag{3.3}$$

By recursively using the above state space equations we can write the input/output equations for the control system

$$U_p = \Gamma^c_N X^c_p + H^c_N(R_p - Y_p) \tag{3.4}$$

$$X^c_f = A^N_c X^c_p + \Delta^c_N(R_p - Y_p) \tag{3.5}$$

$$U_f = \Gamma^c_N X^c_f + H^c_N(R_f - Y_f) \tag{3.6}$$

where

$$\Gamma^c_N = \begin{bmatrix} C_c \\ C_cA_c \\ \dots \\ C_cA_c^{N-1} \end{bmatrix}; \quad H^c_N = \begin{bmatrix} D_c & 0 & \dots & 0 \\ C_cB_c & D_c & \dots & 0 \\ \dots & \dots & \dots & \dots \\ C_cA_c^{N-2}B_c & C_cA_c^{N-3}B_c & \dots & D_c \end{bmatrix}$$

$$\Delta^c_N = \begin{bmatrix} A_c^{N-1}B_c & A_c^{N-2}B_c & \dots & B_c \end{bmatrix};$$

$$X^c_p = \begin{bmatrix} x^c_0 & x^c_1 & \dots & x^c_{j-1} \end{bmatrix};$$

$$X^c_f = \begin{bmatrix} x^c_N & x^c_{N+1} & \dots & x^c_{N+j-1} \end{bmatrix}$$

The matrices R_p and R_f are constructed in the same way as shown in equations (2.4)-(2.7). Using equation (3.4) in equation (3.5) we can derive

$$X^c_f = \begin{bmatrix} A^N_c(\Gamma^c_N)^\dagger & \left(\Delta^c_N - A^N_c(\Gamma^c_N)^\dagger H^c_N\right) \end{bmatrix} \begin{bmatrix} U_p \\ (R_p - Y_p) \end{bmatrix} \tag{3.7}$$

Using equations (2.12) and (3.6) we can derive

$$\begin{bmatrix} Y_f \\ U_f \end{bmatrix} = \begin{bmatrix} (I + H_N H^c_N)^{-1}\Gamma_N & (I + H_N H^c_N)^{-1} H_N \Gamma^c_N \\ -(I + H^c_N H_N)^{-1} H^c_N \Gamma_N & (I + H^c_N H_N)^{-1}\Gamma^c_N \end{bmatrix} \begin{bmatrix} X_f \\ X^c_f \end{bmatrix} + \begin{bmatrix} (I + H_N H^c_N)^{-1} H_N H^c_N \\ (I + H^c_N H_N)^{-1} H^c_N \end{bmatrix} R_f + \begin{bmatrix} (I + H_N H^c_N)^{-1} H^s_N \\ -(I + H^c_N H_N)^{-1} H^c_N H^s_N \end{bmatrix} E_f \tag{3.8}$$

Theorem 3.1 *The input output equations for the closed loop system in equation (3.8) can be equivalently expressed as follows*

$$\begin{bmatrix} Y_f \\ U_f \end{bmatrix} = L_p D_p + \begin{bmatrix} L_{yr}^{CL} \\ L_{ur}^{CL} \end{bmatrix} R_f + \begin{bmatrix} L_{ye}^{CL} \\ L_{ue}^{CL} \end{bmatrix} E_f \tag{3.9}$$

where

$$D_p = \begin{bmatrix} Y_p \\ U_p \\ R_p \\ E_p \end{bmatrix}; \tag{3.10}$$

$$L_{ur}^{CL} = (I + H_N^c H_N)^{-1} H_N^c; \qquad L_{yr}^{CL} = (I + H_N H_N^c)^{-1} H_N H_N^c \tag{3.11}$$

$$L_{ue}^{CL} = -(I + H_N^c H_N)^{-1} H_N^c H_N^s; \qquad L_{ye}^{CL} = (I + H_N H_N^c)^{-1} H_N^s \tag{3.12}$$

Proof: Consider the equation (3.8):

$$\begin{bmatrix} Y_f \\ U_f \end{bmatrix} = \begin{bmatrix} (I + H_N H_N^c)^{-1}\Gamma_N & (I + H_N H_N^c)^{-1} H_N \Gamma_N^c \\ -(I + H_N^c H_N)^{-1} H_N^c \Gamma_N & (I + H_N^c H_N)^{-1}\Gamma_N^c \end{bmatrix} \begin{bmatrix} X_f \\ X_f^c \end{bmatrix} + \begin{bmatrix} (I + H_N H_N^c)^{-1} H_N H_N^c \\ (I + H_N^c H_N)^{-1} H_N^c \end{bmatrix} R_f + \begin{bmatrix} (I + H_N H_N^c)^{-1} H_N^s \\ -(I + H_N^c H_N)^{-1} H_N^c H_N^s \end{bmatrix} E_f$$

Using equation (2.18) and (3.7) in the above expression, we can express

$$\begin{bmatrix} (I + H_N H_N^c)^{-1}\Gamma_N & (I + H_N H_N^c)^{-1} H_N \Gamma_N^c \\ -(I + H_N^c H_N)^{-1} H_N^c \Gamma_N & (I + H_N^c H_N)^{-1}\Gamma_N^c \end{bmatrix} \begin{bmatrix} X_f \\ X_f^c \end{bmatrix} = \begin{bmatrix} L_y^{CL} \\ L_u^{CL} \end{bmatrix} \begin{bmatrix} Y_p \\ U_p \\ R_p \\ E_p \end{bmatrix}$$

where

$$L_y^{CL} = \begin{bmatrix} L_y^1 & L_y^2 & L_y^3 & L_y^4 \end{bmatrix}; \qquad L_u^{CL} = \begin{bmatrix} L_u^1 & L_u^2 & L_u^3 & L_u^4 \end{bmatrix}$$

$$L_y^1 = (I + H_N H_N^c)^{-1}\Gamma_N A^N \Gamma_N^\dagger - (I + H_N H_N^c)^{-1} H_N \Gamma_N^c \left(\Delta_N^c - A_c^N (\Gamma_N^c)^\dagger H_N^c\right)$$

$$L_y^2 = (I + H_N H_N^c)^{-1}\Gamma_N \left(\Delta_N - A^N \Gamma_N^\dagger H_N\right) + (I + H_N H_N^c)^{-1} H_N \Gamma_N^c A_c^N (\Gamma_N^c)^\dagger$$

$$L_y^3 = (I + H_N H_N^c)^{-1} H_N \Gamma_N^c \left(\Delta_N^c - A_c^N (\Gamma_N^c)^\dagger H_N^c\right)$$

$$
\begin{aligned}
L_y^4 &= (I + H_N H_N^c)^{-1}\Gamma_N \left(\Delta_N^s - A^N \Gamma_N^\dagger H_N^s\right) \\
L_u^1 &= -(I + H_N^c H_N)^{-1} H_N^c \Gamma_N A^N \Gamma_N^\dagger - (I + H_N^c H_N)^{-1}\Gamma_N^c \left[\Delta_N^c - A_c^N (\Gamma_N^c)^\dagger H_N^c\right] \\
L_u^2 &= -(I + H_N^c H_i)^{-1} H_N^c \Gamma_N \left(\Delta_N - A^N \Gamma_N^\dagger H_N\right) + (I + H_N^c H_N)^{-1}\Gamma_N^c A_c^N (\Gamma_N^c)^\dagger \\
L_u^3 &= (I + H_N^c H_N)^{-1}\Gamma_N^c \left[\Delta_N^c - A_c^N (\Gamma_N^c)^\dagger H_N^c\right] \\
L_u^4 &= -(I + H_N^c H_N)^{-1} H_N^c \Gamma_N \left(\Delta_N^s - A^N \Gamma_N^\dagger H_N^s\right)
\end{aligned}
$$

Therefore, we can write

$$
\begin{bmatrix} Y_f \\ U_f \end{bmatrix} = L_p D_p + \begin{bmatrix} L_{yr}^{CL} \\ L_{ur}^{CL} \end{bmatrix} R_f + \begin{bmatrix} L_{ye}^{CL} \\ L_{ue}^{CL} \end{bmatrix} E_f \tag{3.13}
$$

With the above theorem, the estimation of the closed loop subspace matrices using the closed loop data is essentially an open loop identification problem. We can define

$$
\begin{bmatrix} Y_f \\ U_f \end{bmatrix} = \begin{bmatrix} L_y^{CL} \\ L_u^{CL} \end{bmatrix} M_p + \begin{bmatrix} L_{yr}^{CL} \\ L_{ur}^{CL} \end{bmatrix} R_f + \begin{bmatrix} L_{ye}^{CL} \\ L_{ue}^{CL} \end{bmatrix} E_f
$$

where $M_p = \begin{bmatrix} Y_p \\ U_p \\ R_p \end{bmatrix}$. From the above equation, and since R_f can be chosen as a random binary signal uncorrelated with M_p and E_f, the closed loop subspace matrices $\{L_u^{CL}, L_{ur}^{CL}, L_y^{CL}, L_{yr}^{CL}\}$ can be obtained as a solution of the least squares estimation problem. $\{\hat{U}_f$ and $\hat{Y}_f\}$ are found by the orthogonal projection of the row space of $\{U_f$ and $Y_f\}$ into the row space spanned by $\{M_p$ and $R_f\}$.

$$
\begin{aligned}
\begin{bmatrix} L_u^{CL} & L_{ur}^{CL} \end{bmatrix} &= U_f \begin{bmatrix} M_p \\ R_f \end{bmatrix}^\dagger \\
&= U_f \begin{bmatrix} M_p^T & R_f^T \end{bmatrix} \left(\begin{bmatrix} M_p \\ R_f \end{bmatrix} \begin{bmatrix} M_p^T & R_f^T \end{bmatrix} \right)^{-1} \qquad (3.14) \\
\begin{bmatrix} L_y^{CL} & L_{yr}^{CL} \end{bmatrix} &= Y_f \begin{bmatrix} M_p \\ R_f \end{bmatrix}^\dagger
\end{aligned}
$$

$$= Y_f \begin{bmatrix} M_p^T & R_f^T \end{bmatrix} \left(\begin{bmatrix} M_p \\ R_f \end{bmatrix} \begin{bmatrix} M_p^T & R_f^T \end{bmatrix} \right)^{-1} \tag{3.15}$$

This projection can be implemented in a numerically robust way with a QR-decomposition. The first row of $\hat{Y}_f$ represents the one-step ahead predictions for the input. Therefore the white noise disturbance sequence entering the process can be estimated as

$$\hat{e}_f = \begin{bmatrix} \hat{e}_N & \hat{e}_{N+1} & \dots & \hat{e}_{N+j-1} \end{bmatrix}^T = Y_f(1:m,:) - \hat{Y}_f(1:m,:) \tag{3.16}$$

where $\bullet(1:m,:)$ represents a vector containing the elements from rows 1 to m and all the columns of the matrix. Let us define

$$\Xi_f = U_f - \hat{U}_f = L_{ue}^{CL} E_f \tag{3.17}$$

The block Hankel matrix, E_f, for the noise can be constructed using the estimated noise, e_f. Therefore, L_{ue}^{CL} is estimated as,

$$L_{ue}^{CL} = \Xi_f / E_f = \Xi_f E_f^{\dagger} \tag{3.18}$$

3.2.1 Estimation of the process model

Consider the following identities:

$$(\mathcal{AB})^{-1} = \mathcal{B}^{-1}\mathcal{A}^{-1}; \qquad \mathcal{A}^{-1}\mathcal{B} = \mathcal{A}^{-1}(\mathcal{B}^{-1})^{-1} = (\mathcal{B}^{-1}\mathcal{A})^{-1}$$

we can write

$$\begin{aligned} L_{yr}^{CL} &= (I + H_N H_N^c)^{-1} H_N H_N^c = \{(H_N H_N^c)^{-1}(I + H_N H_N^c)\}^{-1} \\ &= \{(H_N H_N^c)^{-1} + I\}^{-1} \end{aligned} \tag{3.19}$$

$$\begin{aligned} L_{ur}^{CL} &= (I + H_N^c H_N)^{-1} H_N^c = \{(H_N^c)^{-1}(I + H_N^c H_N)\}^{-1} \\ &= \{(H_N^c)^{-1} + H_N\}^{-1} \end{aligned} \tag{3.20}$$

Therefore,

$$\begin{aligned} L_{yr}^{CL}(L_{ur}^{CL})^{-1} &= \{(H_N H_N^c)^{-1} + I\}^{-1} \{(H_N^c)^{-1} + H_N\} \\ &= \{[(H_N^c)^{-1} + H_N]^{-1}[(H_N H_N^c)^{-1} + I]\}^{-1} \end{aligned}$$

$$= \{[(H_N^c)^{-1} + H_N]^{-1}[(H_N^c)^{-1} + H_N]H_N^{-1}\}^{-1}$$
$$= (H_N^{-1})^{-1} = H_N \tag{3.21}$$

Hence H_N, which contains the Markov parameters corresponding to the deterministic input, can be identified as

$$H_N = L_{yr}^{CL}(L_{ur}^{CL})^{-1} = \begin{bmatrix} G_0 & 0 & \dots & 0 \\ G_1 & G_0 & \dots & 0 \\ \dots & \dots & \dots & 0 \\ G_{N-1} & \dots & \dots & 0 \end{bmatrix} \tag{3.22}$$

where G_i represents the Markov parameter of the deterministic input of the i-th delay. The dynamic matrix containing the system step response coefficients, S_N, can be obtained as

$$S_N = \begin{bmatrix} s_0 & 0 & \dots & 0 \\ s_1 & s_0 & \dots & 0 \\ \dots & \dots & \dots & 0 \\ s_{N-1} & \dots & \dots & 0 \end{bmatrix} = \begin{bmatrix} G_0 & 0 & \dots & 0 \\ G_1 & G_0 & \dots & 0 \\ \dots & \dots & \dots & 0 \\ G_{N-1} & \dots & \dots & 0 \end{bmatrix} \begin{bmatrix} I & 0 & \dots & 0 \\ I & I & \dots & 0 \\ \dots & \dots & \dots & 0 \\ I & I & \dots & I \end{bmatrix} \tag{3.23}$$

$$= H_N \begin{bmatrix} I & 0 & \dots & 0 \\ I & I & \dots & 0 \\ \dots & \dots & \dots & 0 \\ I & I & \dots & I \end{bmatrix} \tag{3.24}$$

where s_i represents the i^{th} step response parameter of the deterministic input.

3.2.2 Estimation of the noise model

The noise model can be estimated from the residuals of the input data. Using the definitions for L_{ur}^{CL} and L_{ue}^{CL} provided in theorem-1, the noise dynamic matrix is obtained as

$$H_N^s = -(L_{ur}^{CL})^{-1}L_{ue}^{CL} \tag{3.25}$$

H_N^s contains the impulse response coefficients corresponding to the stochastic input.

$$H_N^s = \begin{bmatrix} I & 0 & ... & 0 \\ L_1 & I & ... & 0 \\ ... & ... & ... & 0 \\ L_{N-1} & ... & ... & 0 \end{bmatrix} \tag{3.26}$$

where L_i represents the i^{th} impulse response coefficient / Markov parameter of the stochastic inputs. Thus the first column of H_N^s represents the noise model $H(z^{-1})$ in IR-form.

$$H(z^{-1}) = I + L_1 z^{-1} + L_2 z^{-2} + ... + L_i z^{-i} + ... + L_{N-1} z^{-N+1} \tag{3.27}$$

3.3 The algorithm

The following are the steps in the proposed closed loop identification:

Step I. Construction of the data Hankel matrices $\{U_p, U_f, Y_p, Y_f, R_p, R_f\}$ using the closed loop data. By linear regression the deterministic closed loop subspace matrices are identified.

Remarks: The guidelines presented in section 3.4 can be used in the selection of the number of rows and columns. By adding a persistent exciting signal, which is uncorrelated with the process noise, in the setpoint, we ensure unbiased estimation of the closed loop subspace matrices. This step is an open loop identification problem with the setpoint change being the external inputs and the closed-loop subspace matrices as the model to be identified.

Step II. Estimation of the vector of noise data from the 'output data Hankel matrix', and 'residual data Hankel matrix' corresponding to 'input data Hankel matrix'. Estimation of the stochastic closed loop subspace matrices.

Remarks: The first row of the residual matrix $(Y_f - \hat{Y}_f)$ represents one-step ahead prediction errors and an unbiased estimate of the noise entering the process since the feedback does not effect the current noise. The matrix, $\Xi_f = (U_f - \hat{U}_f)$ is the

residual data Hankel matrix corresponding to the input. The noise data Hankel matrix, E_f, is constructed using the vector of estimated process noise. By linear regression, the stochastic closed loop subspace matrix, L_e^I, is estimated.

Step III. Retrieving the open loop deterministic subspace matrix from the closed loop subspace matrices.

Remarks: Closed loop subspace matrices are just the open loop subspace matrices weighted by the subspace matrix corresponding to the sensitivity function. The analogies between the process/noise transfer functions and the open loop deterministic/stochastic subspace matrices are obvious. The method presented in this chapter is parallel to the approach used in 'joint input/output closed loop identification method', which is well known in the transfer function domain. However with the 'joint input/output closed loop identification method', inverting the transfer function (or transfer function matrices for the multivariate systems) can give problems such as, the resultant transfer function (matrix) may be improper or of high order. No such problems are encountered in the subspace matrices based approach proposed in this chapter since we are dealing with matrices instead of transfer functions, provided the closed loop subspace matrices are of full rank. See the guidelines in section 3.4 for avoiding the closed loop subspace matrices from becoming rank deficient.

Step IV. Retrieving the open loop stochastic subspace matrix from the closed loop subspace matrices.

3.4 Some guidelines for the practical implementation of the algorithm

Building the data Hankel matrices is the first step in all the subspace based identification methods. If one wants to identify a state space model for the system using the subspace identification methods, then the number of rows, N, is chosen to be higher than the order of the state space model to be identified [93]. The number

of columns, j, should tend to infinity. Since we have finite data in real situations we can only choose a finite j, the maximum number of columns that can be constructed with the available data. As far as the closed loop identification method presented in this chapter is concerned, we are identifying only the subspace matrices and *not* the state space system matrices. Here are some guidelines that can help in deciding the number of rows and columns of the data Hankel matrices:

a. To obtain the complete process model, the number of rows, N, should be chosen such that the last impulse response coefficient (last element of the first block-column of H_N) is close to zero.

b. The choice of the number of columns really depends on the excitation signal used for identification. The richer the excitation signal the fewer the number of columns required. The number of columns is chosen in such a way that the corresponding impulse response coefficients in the columns of the subspace matrices are very close. The higher the number of columns taken in the data Hankel matrices the closer will the corresponding coefficients in the columns be.

c. It may be a good idea to check the rank of the closed loop subspace matrices before retrieving the open loop subspace matrices. If the closed loop subspace matrices are rank deficient then either decrease the number of rows or increase the number of columns of the data Hankel matrices. However decreasing the number of rows decreases the number of Markov parameters obtained in the identified subspace matrices.

d. Numerical tools like QR-decomposition can be used to avoid numerical problems associated with the inversion of large matrices, specially in step 1.

e. Multivariate systems: Although, in principle, all the derivations in this chapter are applicable to multivariate systems, numerical problem is a potential concern. As the number of variables increases the size of the data Hankel matrices can be prohibitively high, specially for systems with a long settling time. MISO identification instead of MIMO identification can reduce the size of the data Hankel matrices. Slower sampling can be used for processes with slow dynamics. Numerical

techniques such as QR-decomposition can be used to deal with the inversion of large matrices for multivariate systems.

f. Studies on derivation of statistical properties for subspace based identification methods are an area of active research and have been considered in [53, 67, 93, 119] and the references therein, where it has been shown that under open-loop condition subspace identification can yield a consistent estimation of the parameters. As the proposed method in this chapter is equivalent to an open-loop subspace identification problem, the same conclusion can be applied.

3.5 Extension to the case of measured disturbance variables

The closed loop subspace based identification method explained in the previous section can be extended to the case where some measured disturbances are available for feedforward control. Consider the case when measurements of some of the disturbance variables are available and we want to identify the subspace matrix corresponding to these variables. Let $v_t(h \times 1)$ represents the vector of measured disturbance variables. Assume that the measured disturbance variables are uncorrelated with the setpoint changes. Consider a feedback-only controller described in equations (3.1)-(3.6) acting on the process represented by

$$x_{k+1} = Ax_k + \begin{bmatrix} B & B_v \end{bmatrix} \begin{bmatrix} u_k \\ v_k \end{bmatrix} + K^f e_k \tag{3.28}$$

$$y_k = Cx_k + \begin{bmatrix} D & D_v \end{bmatrix} \begin{bmatrix} u_k \\ v_k \end{bmatrix} + e_k \tag{3.29}$$

The matrix input-output equations (2.12) and (2.19) are modified to include measured disturbances

$$Y_f = \Gamma_N X_f^b + H_N U_f + H_N^v V_f + H_N^s E_f \tag{3.30}$$

$$= L_w^b W_p^b + L_u U_f + L_v V_f + L_e E_f \tag{3.31}$$

where

$$H_N^v = \begin{bmatrix} D^v & 0 & ... & 0 \\ CB^v & D^v & ... & 0 \\ ... & ... & ... & ... \\ CA^{N-2}B^v & CA^{N-3}B^v & ... & D^v \end{bmatrix} \tag{3.32}$$

$$W_p^b = \begin{bmatrix} Y_p \\ U_p \\ V_p \end{bmatrix} \tag{3.33}$$

and, V_p and V_f are the 'past' and 'future' data Hankel matrices of v_t defined in the same way as those corresponding to u_t defined in equations (2.4)-(2.5).

Similar to equation (3.14) we can derive

$$\begin{bmatrix} Y_f \\ U_f \end{bmatrix} = \begin{bmatrix} L_y^{CL} \\ L_u^{CL} \end{bmatrix} M_p + \begin{bmatrix} L_{yr}^{CL} \\ L_{ur}^{CL} \end{bmatrix} R_f + \begin{bmatrix} L_{yv}^{CL} \\ L_{uv}^{CL} \end{bmatrix} V_f + \begin{bmatrix} L_{ye}^{CL} \\ L_{ue}^{CL} \end{bmatrix} E_f$$

where in addition to the definitions in equations (3.11)-(3.12) we used

$$L_{uv}^{CL} = -(I + H_N^c H_N)^{-1} H_N^c H_N^v; \qquad L_{yv}^{CL} = (I + H_N H_N^c)^{-1} H_N^v \tag{3.34}$$

The closed loop subspace matrices are identified by data projections as shown in section 3.2. The matrix H_N^v containing the Markov parameters corresponding to the measured disturbance variables is obtained from the closed loop subspace matrices with

$$H_N^v = -(L_{ur}^{CL})^{-1} L_{uv}^{CL} \tag{3.35}$$

3.6 Closed loop simulations

Certain comparative simulations were carried out in MATLAB for two cases, univariate and multivariate systems, between the non-parametric approach presented in this chapter, MOESP [122] and CVA [71, 113]. The purpose of this exercise is to check the validity of the proposed non-parametric approach

of closed loop identification for both univariate and multivariate systems, which does not require controller knowledge for closed loop identification, and also to see how it performs compared to the existing subspace identification methods, the MOESP/CVA approaches[2].

3.6.1 Univariate system

Consider the following system [93]:

$$
\begin{aligned}
x_{k+1} &= \begin{bmatrix} 0.6 & 0.6 & 0 \\ -0.6 & 0.6 & 0 \\ 0 & 0 & 0.7 \end{bmatrix} x_k + \begin{bmatrix} 1.6161 \\ -0.3481 \\ 2.6319 \end{bmatrix} u_k + \begin{bmatrix} -1.1472 \\ -1.5204 \\ -3.1993 \end{bmatrix} e_k \\
y_k &= \begin{bmatrix} -0.4373 & -0.5046 & 0.0936 \end{bmatrix} x_k + [-0.7759] u_k + e_k
\end{aligned}
$$

where x_k, y_k, u_k and e_k represent the system state, output, input and the unmeasured random noise respectively at time k. A PID controller, $0.1 + \frac{0.08}{s} + 0.08s$, is tuned online for the above system for good setpoint tracking and disturbance rejection performance. We assume that the controller knowledge is unknown for the closed loop identification. 'Close loop input/output/setpoint' data is obtained by exciting the system using a designed 'RBS' signal of magnitude 1 for the system output setpoint and random white noise of standard deviation 0.1 in MATLAB-*Simulink*. The closed loop data is plotted in figure (3.3). Using the closed loop subspace identification method presented in section 3.2, with rows(N) = 30 and columns(j) = 2000 in the data Hankel matrices, the subspace matrices H_N and H_N^s are identified. Due to the presence of noise, the upper non-diagonal elements in H_N and H_N^s will not be exactly zero but very small numbers (they approach zero as $j \longrightarrow \infty$). The true impulse response coefficients of the system can be calculated from the state

[2]It has been shown by Van Overschee and De Moor [92] that the difference between the three subspace identification algorithms N4SID/MOESP/CVA is the difference in the way the weighting matrices are used in the subspace identification algorithm. In fact, MATLAB-6 offers a feature called *N4weight* for the 'N4SID' command wherein the user can specify MOESP or CVA and the respective weighting matrices will be used, so that it is equivalent to using MOESP/CVA subspace identification algorithms. Hence in the simulations presented in this section, the MOESP/CVA weighting matrices are used in the 'N4SID' algorithm in MATLAB, instead of writing separate algorithms for MOESP/CVA.

space system matrices provided above. The identified impulse response coefficients are plotted against the true impulse response coefficients in figure (3.4). It is illustrated that the identified impulse response coefficients match very well with the true coefficients. Closed loop MOESP/CVA is used to identify the deterministic part of the system using the same set of closed loop data. Impulse response coefficients identified through MOESP and CVA are plotted against the true impulse response coefficients in figure (3.5). We can see that impulse response coefficients identified from both MOESP and CVA methods are reasonably well matching with the true impulse response coefficients in this univariate systems. Therefore, all three methods yield similar results for this univariate systems.

3.6.2 Multivariate system

Consider the following system taken from MATLAB/MPC toolbox manual.

$$\begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix} = \begin{bmatrix} \frac{12.8e^{-s}}{16.7s+1} & \frac{-18.9e^{-3s}}{21.0s+1} \\ \frac{6.6e^{-7s}}{10.9s+1} & \frac{-19.4e^{-3s}}{14.4s+1} \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix} + \begin{bmatrix} \frac{3.8e^{-8s}}{14.9s+1} \\ \frac{4.9e^{-3s}}{13.2s+1} \end{bmatrix} w(s)$$

where $\{y_1(s), y_2(s)\}$, $\{u_1(s), u_2(s)\}$ and $w(s)$ represent the system outputs, inputs and random noise disturbance respectively. A state space based MPC controller is designed in MATLAB for the system. A sampling period of $T = 2$ time units is used in the simulations. Close loop input/output data is obtained by exciting the system using a designed 'RBS' signal of magnitude 1 for the setpoint (r_t) and random white noise (w_t) of standard deviation 0.1 in MATLAB-*Simulink*. We assume that the controller and w_t are not known for the closed loop identification. The closed loop data is plotted in figure (3.6). The proposed non-parametric closed loop identification algorithm is used with rows(N) = 50 and columns(j) = 2500 in the data Hankel matrices to identify the subspace matrices H_N and H_N^s. The identified impulse response coefficients are plotted against the true impulse response coefficients in figure (3.7). It can be see from the plot that the identified impulse response coefficients match very closely with the true coefficients.

Next, the MOESP approach is used to identify the deterministic part of the system

using closed loop data. The impulse response coefficients identified using MOESP approach are compared with the true coefficients in figure (3.8) and the result is not quite comparable with the one obtained using the proposed method. It should be noted that a better match between the identified and true coefficients using MOESP could be achieved only with a very high order model (the resultant deterministic model being at least of 20th order or more). The order of the end model could probably be reduced using a standard model reduction method but with a compromise in terms of bias error and complexity. The CVA approach [71] involves the inversion of a transfer function matrix which may not always be possible in MATLAB due to the time delays or due to the fact that the resultant matrix could contain improper transfer functions.

3.7 Identification of the dynamic matrix: Practical application on a pilot scale plant

The proposed method for the estimation of the dynamic matrix from closed loop data is tested on a pilot scale system. The system considered is shown in figure (3.9). The input (u) is the inlet water flow rate and the process variable to be controlled (y) is the level of water in the tank. The tank outlet flow valve is kept at a constant position. The head of the water in the inlet pipe can be considered as (an unmeasured) disturbance. The tank level is controlled by a PID controller, $2.5+\frac{0.05}{s}+1s$. An 'RBS' signal of series of setpoint changes to the level is designed in $MATLAB$. Closed loop data of the process input, setpoint and output is collected and plotted in figure (3.10).

Data Hankel matrices of dimensions rows(N) = 200 and columns(j) = 1500 are constructed for the closed loop data and the subspace matrices H_N and H_N^s are identified using the closed loop identification method presented in the previous sections. The columns of the subspace matrices are plotted in figure (3.11). It is illustrated in the figure that the impulse response coefficients in the columns of

H_i are matching with each other.

The accuracy of the impulse response coefficients in the matrix H_N is checked by doing an open loop identification. Open loop data is collected by exciting the process with an input 'rbs' signal of magnitude 1. The impulse response coefficients identified using the open loop subspace identification method are plotted together with the coefficients identified suing closed loop data in figure (3.12).

We can see that there is some mismatch in the impulse response models in the subspace matrices identified using closed loop data and those identified using open loop data. The mismatch may be due to the different operating regions excited between closed loop and open loop identifications and effect of feedback control. The noise model mismatch may be traced to the time varying nature of the disturbances entering the process.

3.8 Conclusions

This chapter provides a subspace identification based method for the identification of process dynamic matrix and the noise model from closed loop data. The closed loop subspace matrices are first obtained by persistent setpoint excitation of the closed loop system. The open loop subspace matrices are then retrieved from the closed loop subspace matrices. The process dynamic matrix is obtained from the deterministic subspace matrix and the noise model in the impulse response form is obtained from the stochastic subspace matrix. The method can be easily extended to the case of measured disturbances. Results from computer simulations and practical application on a pilot scale plant are provided to illustrate the proposed closed loop identification method.

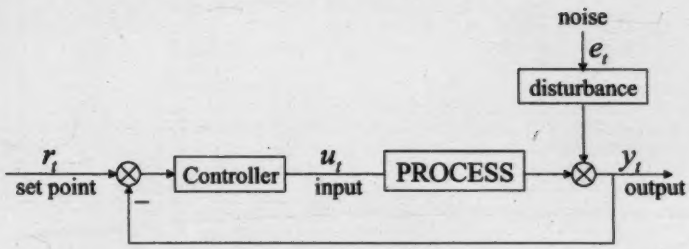


Figure 3.1: Closed loop system.

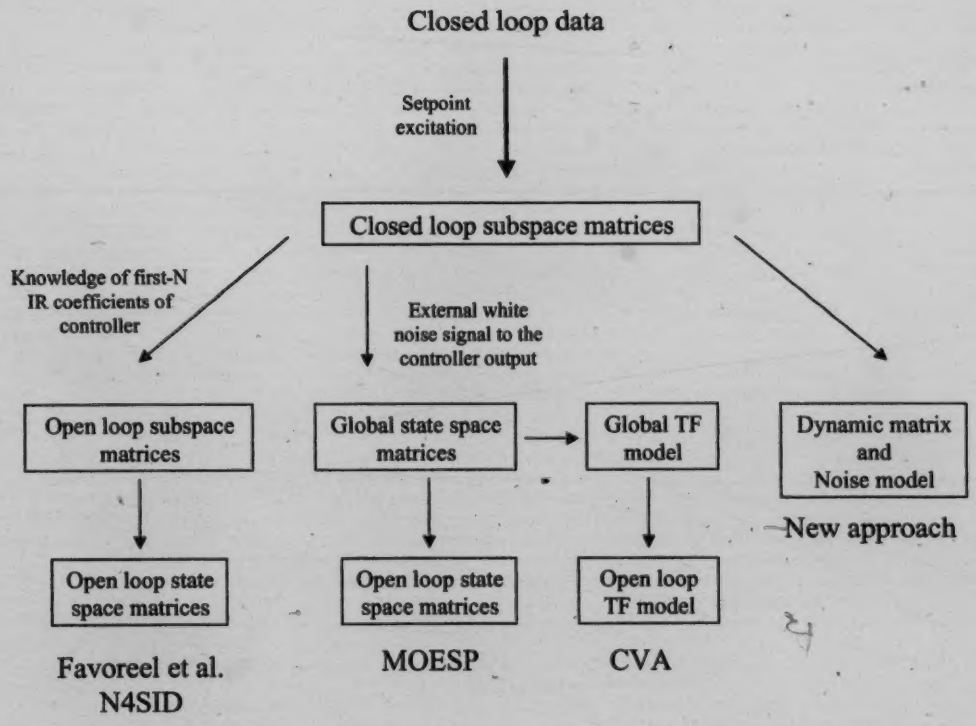


Figure 3.2: Comparing the existing closed loop subspace state space identification methods and the new approach.

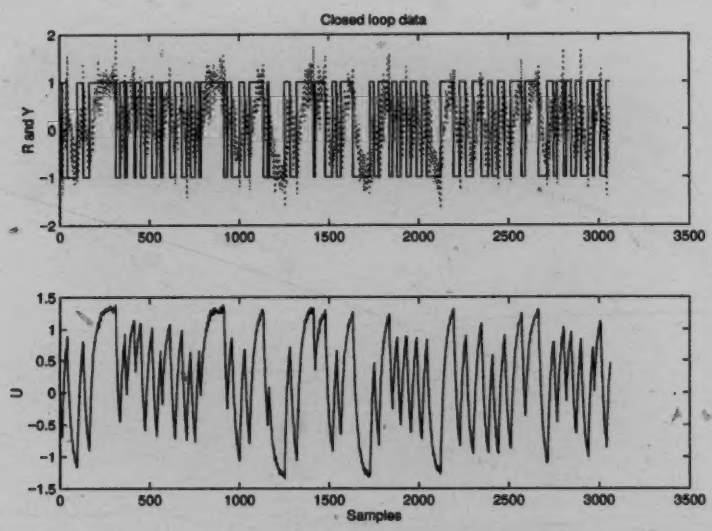


Figure 3.3: univariate system: Closed loop system data

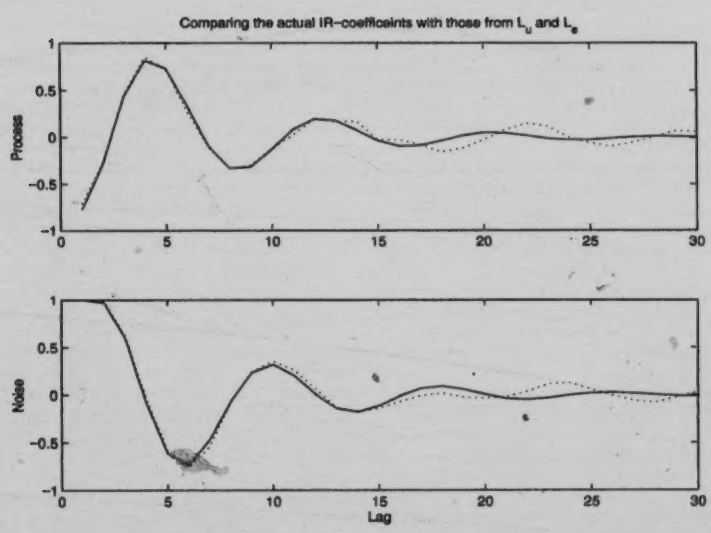


Figure 3.4: univariate system: Comparing the true (solid) IR-coefficients with those obtained in the subspace matrices (dotted).

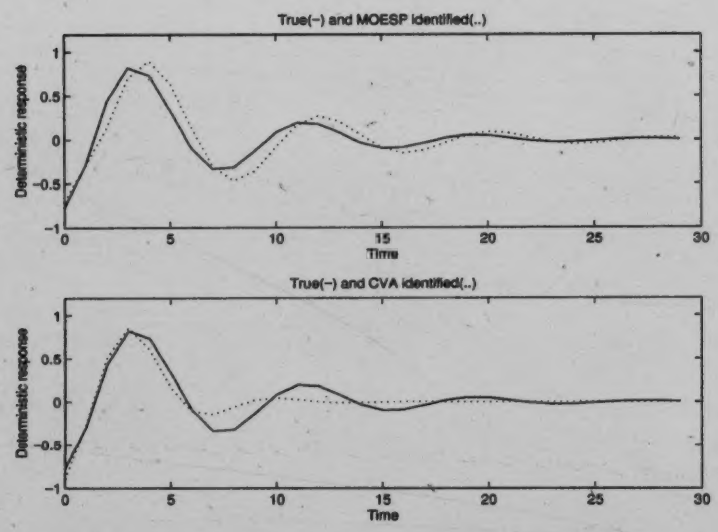


Figure 3.5: univariate system: Comparing the true (solid) IR-coefficients with those identified by MOESP/CVA approaches.

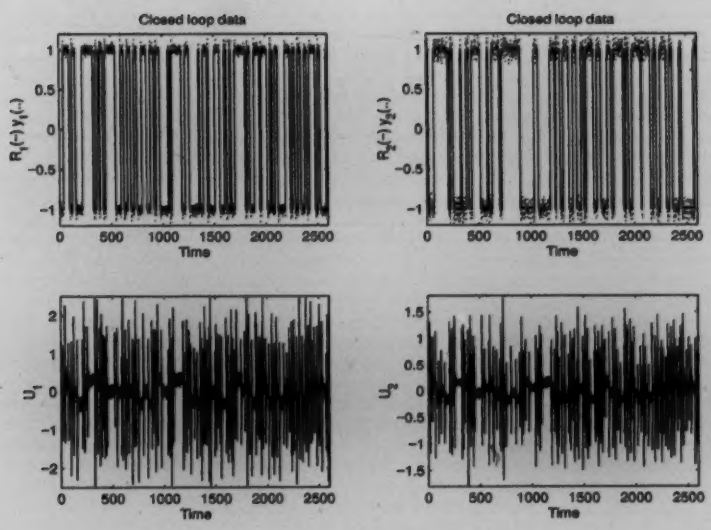


Figure 3.6: Multivariate system: Closed loop system data

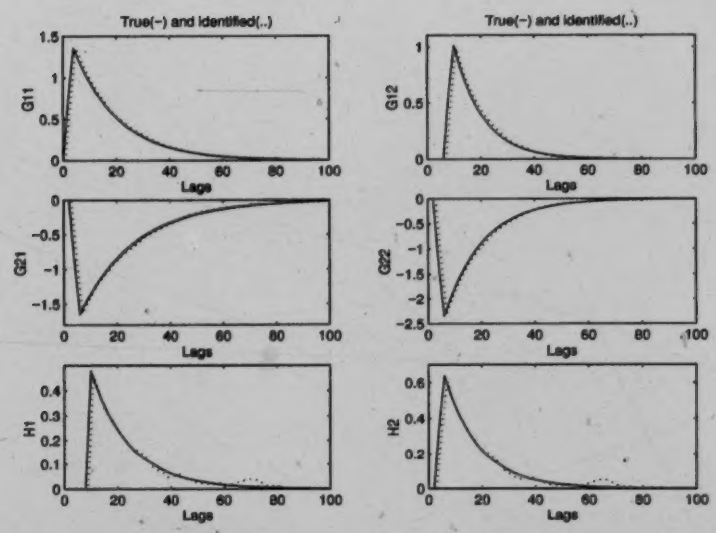


Figure 3.7: Multivariate system: Comparing the true (solid) IR-coefficients with those obtained in the subspace matrices (dotted).

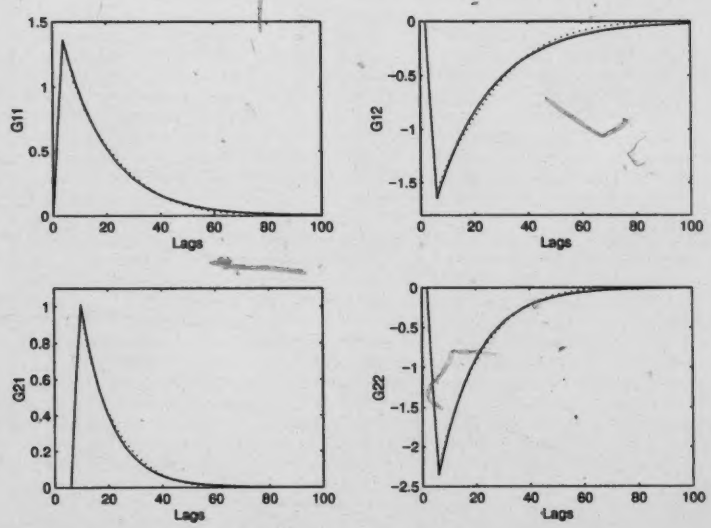


Figure 3.8: Multivariate system: Comparing the true (solid) IR-coefficients with those identified by MOESP approach.

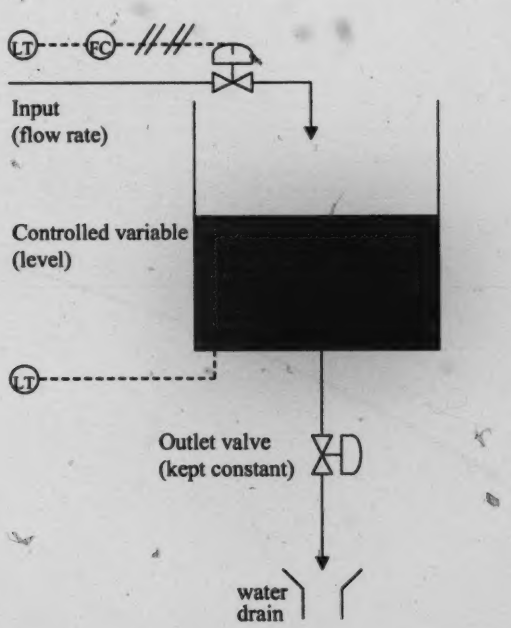


Figure 3.9: Experimental setup.

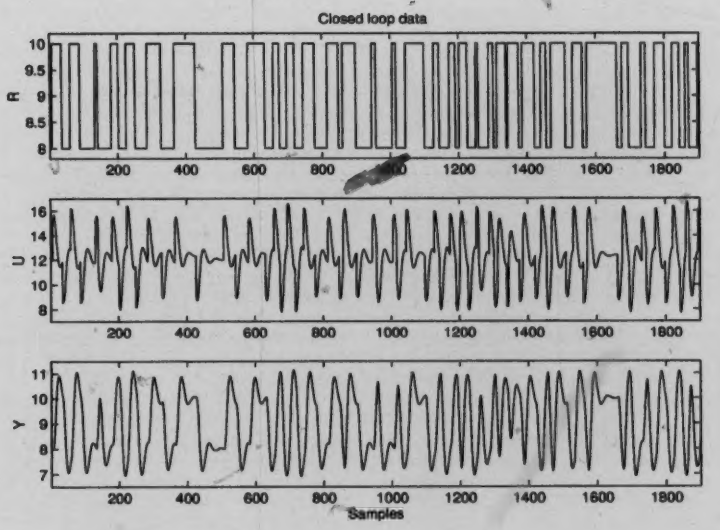


Figure 3.10: Pilot scale process: Closed loop system data.

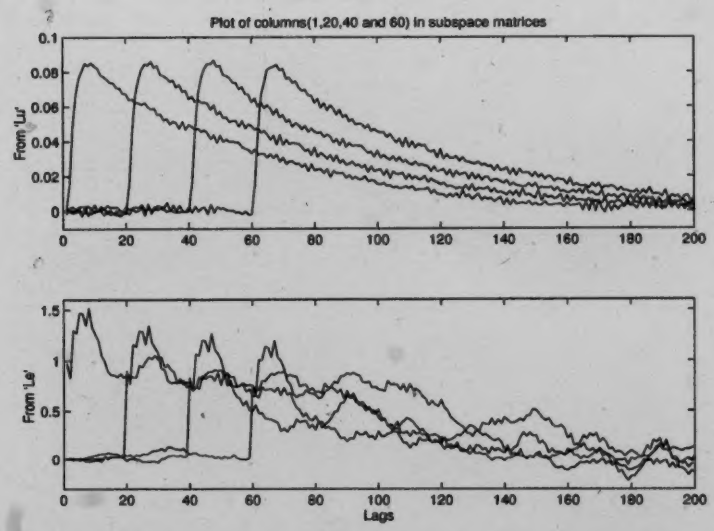


Figure 3.11: Pilot scale process: IR-coefficients from the consecutive columns of the identified subspace matrices.

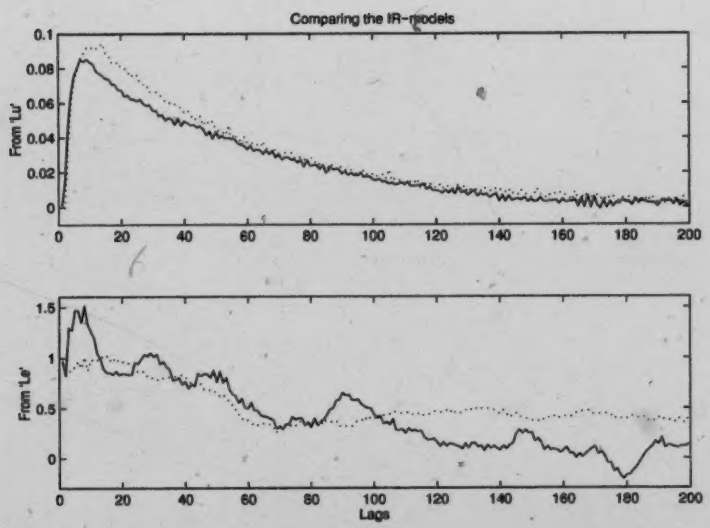


Figure 3.12: Pilot scale process: Comparing the IR-coefficients from subspace matrices identified using the open loop data (dotted line) and that from the closed loop data (solid line).

Chapter 4

A data driven subspace approach to predictive controller design

[1]

4.1 Introduction

Predictive controllers have been widely used in process industries for more than two decades [10, 78, 88, 99]. Several forms of predictive controllers like IDCOM [101], DMC [15, 16, 78], QDMC [14, 30], GPC [12, 13], etc. have been proposed and successfully implemented in process industries through the years. The term predictive control does not designate a specific control strategy but a wide range of control algorithms which make an explicit use of a process model in a cost function minimization to obtain the control signal [10, 29]. Hence a model of the process is the basic requirement for the design of predictive controllers; this is first identified using plant input and output data. From the process model, predictor matrices can be obtained (for example, the dynamic matrix constructed

[1] A version of this chapter has been published as a journal paper

R. Kadali, B. Huang, and A. Rossiter. A data driven subspace approach to predictive controller design. *Control Eng. Practice*, 11(3); 261-278, 2003.

using step response coefficients in DMC [15, 16]). The predictor matrices are used to obtain predictions of the process output which are used in the controller design. However, it has been found recently that these predictor matrices can be directly obtained from the input/output data by using the subspace matrices (a term used in the subspace identification literature), eliminating the intermediate step of process model identification and providing a means for designing a predictive controller, in the generalized predictive controller (GPC) framework (e.g. [56]), without a parametric model. Since no traditional parametric model is required for the controller design this approach is also referred to as the "model-free approach", and this term has been adopted in the literature (for example see [26, 112]). The idea is to obtain the controller matrices used in the predictive controllers directly from the data without the intermediate parametric model identification step. Hence this approach can also be considered as a direct data driven approach. Moreover, subspace identification methods involve minimizing the summation of multi-step ahead prediction errors, making the subspace matrices based design approach a suitable approach for predictive control.

The predictive controller based on subspace matrices uses the same cost-function as GPC and hence an important question is how one obtains the predictions utilized within the cost function. One of the key aspects in GPC is the assumption of an ARIMAX model for the process [7, 12]. This requires pre-specification of the order and structure of the model to be identified for controller design. Typically one uses reduced complexity models which frequently introduces bias errors. In the traditional (prediction error methood) PEM approach, the model is usually identified in a nonlinear, iterative manner, and in general Diophantine equations need to be solved to obtain the prediction matrices. On the other hand, the predictive controller designed using subspace matrices makes no pre-assumptions about the structure and order of the process model (alleviating some bias errors). Moreover the prediction matrices are obtained through a single matrix algebraic calculation. In summary, the subspace approach to predictive control has the key features of GPC [12, 13] like: (1) long-range prediction over a finite horizon; (2)

inclusion of weighting on outputs and control moves in the cost-function and (3) choice of a prediction horizon and a control horizon after which projected control moves are taken to be zero. It combines these with the added advantages of: (1) no pre-assumptions about model order or structure; (2) parametric matrices obtained in a single iteration and (3) not having to solve Diophantine equations. We also note that extension to the multivariate systems is straightforward with the subspace approach.

Although the idea of designing predictive controllers using the subspace matrices, such as model-free LQG and subspace predictive controller [23, 25, 26], or using the state space model identified through subspace approach [103, 104, 106], has been around for a few years, designing a predictive controller from subspace matrices with all the features of the traditional predictive controller has not been investigated fully. The equivalence of finite horizon LQG to GPC is well known [7]; however there are several other important issues that need to be addressed in the subspace predictive control framework and they form the main contribution of this chapter. The following are the issues considered in this chapter: (1) derivation of a predictive control law in the GPC framework (with systematic inclusion of integral action, an issue ignored in previous works); (2) extension of the predictive control law to include feedforward control to compensate for measured disturbances; (3) inclusion of a constraint handling facility and (4) tuning of the noise model.

The chapter is arranged as follows. Section 4.2 gives an overview of GPC design. Subspace approach to the predictive controller design with enhanced features is explained in section 4.3. Inclusion of the independent noise model for tuning is discussed in section 4.4. Results from the simulation and actual implementation on a pilot scale plant using the proposed predictive control scheme are presented in section 4.5 and section 4.6 respectively. The conclusions are presented in section 4.7.

4.2 Revisit of GPC

GPC design [7, 12, 13] starts by first identifying an ARIMAX model for the process, expressed as

$$A(z^{-1})y_t = B(z^{-1})u_{t-1} + \frac{C(z^{-1})}{\Delta}e_t \tag{4.1}$$

A, B and C are polynomials in the backshift operator, z^{-1}, with A and C being monic. $\Delta = (1 - z^{-1})$ is the differencing operator. The role of the Δ is to ensure integral action in the controller by including an internal disturbance model of typical load perturbations arising in the process industry [7]. A popular quadratic cost function to be minimized is

$$J = \sum_{k=N_1}^{N_2}(r_{t+k} - \hat{y}_{t+k|t})^2 + \sum_{k=1}^{N_u}\lambda(\Delta u_{t+k-1})^2 \tag{4.2}$$

with N_2 and N_u being the prediction and control horizons respectively and λ being the weighting on the control effort. N_1 is usually chosen as 1 or the process time delay t_d. r_{t+k} is the future setpoint for time instant $t + k$. For a discussion on the selection of values for N_1, N_2, N_u and λ, readers are referred to [12] and [13]. Using the Diophantine equations

$$\frac{C(z^{-1})}{A(z^{-1})\Delta} = E_k + q^{-k}\frac{F_k}{A(z^{-1})\Delta} \tag{4.3}$$

$$E_k B = G_k C + q^{-k}\Gamma_k \tag{4.4}$$

and equations (1,2) (and ignoring the term $E_k e_{t+k}$) we obtain the k-step ahead output prediction equation

$$\begin{aligned}\hat{y}_{t+k} &= \frac{F_k}{C}y_t + \frac{\Gamma_k}{C}\Delta u_{t-1} + G_k\Delta u_{t+k-1} \\ &= F_k y_t^f + \Gamma_k \Delta u_{t-1}^f + G_k \Delta u_{t+k-1} \\ &= f(k) + G_k \Delta u_{t+k-1}\end{aligned} \tag{4.5}$$

where

$$\Delta u_t^f = C^{-1}\Delta u_t; \quad y_t^f = C^{-1}y_t$$

and $f(k) = F_k y_t^f + \Gamma_k u_{t-1}^f$ is the free response of the process. Define the vectors of predictions

$$\triangle u_f = \left[\begin{array}{cccccc} \triangle u_t & \ldots & \triangle u_{t+k} & \ldots & \triangle u_{t+N_u} \end{array} \right]^T; \quad \hat{y}_f = \left[\begin{array}{cccccc} \hat{y}_{t+1} & \ldots & \hat{y}_{t+k} & \ldots & \hat{y}_{t+N_2} \end{array} \right]^T$$

$$r_f = \left[\begin{array}{cccccc} r_{t+1} & \ldots & r_{t+k} & \ldots & r_{t+N_2} \end{array} \right]^T; \quad F_f = \left[\begin{array}{cccccc} f(1) & \ldots & f(k) & \ldots & f(N_2) \end{array} \right]^T$$

then the multi step predictor equations can be expressed as

$$\hat{y}_f = G\triangle u_f + F_f \tag{4.6}$$

where G is the dynamic matrix containing the step response coefficients of $\frac{B}{A}$ or the impulse response coefficients of $\frac{B}{A\triangle}$. The GPC control law becomes

$$\triangle u_f = (G^T G + \lambda I)^{-1} G^T (r_f - F_f) \tag{4.7}$$

4.3 Predictive controller design from subspace matrices

Consider a controller objective function which is the same as that of GPC. To simplify the notation, assume $N_1 = 1$. The cost function to be minimized becomes:

$$\begin{aligned} J &= \sum_{k=1}^{N_2} (\hat{y}_{t+k|t} - r_{t+k})^2 + \sum_{k=1}^{N_u} \lambda (\triangle u_{t+k-1})^2 \\ &= (r_f - \hat{y}_f)^T (r_f - \hat{y}_f) + \triangle u_f^T (\lambda I) \triangle u_f \end{aligned} \tag{4.8}$$

where the future outputs are over the prediction horizon, $t+1$ to $t+N_2$, and the future incremental inputs are over the control horizon, t to $t+N_u-1$. For the state space representation (2.1)-(2.2), the vector of the optimal prediction of the future outputs can be expressed in terms of the future inputs and current states as

$$\hat{y}_f = \left[\begin{array}{cccccc} \hat{y}_{t+1} & \ldots & \hat{y}_{t+N_2} \end{array} \right]^T \tag{4.9}$$

$$= \left[\begin{array}{c} C \\ CA \\ \ldots \\ CA^{N_2-1} \end{array} \right] x_t + \left[\begin{array}{cccc} D & 0 & 0 & \ldots \\ CB & D & 0 & \ldots \\ \ldots & \ldots & \ldots & \ldots \\ CA^{N_2-2}B & \ldots & D & \ldots \end{array} \right] \left[\begin{array}{c} u_t \\ u_{t+1} \\ \ldots \\ u_{t+N_u-1} \end{array} \right] \tag{4.10}$$

$$= \Gamma_{N_2} x_t + H(1:N_2, 1:N_u) u_f \tag{4.11}$$

$$= L_w w_p + L_u u_f \tag{4.12}$$

where $w_p = \begin{bmatrix} y_{t-N+1} & \cdots & y_t & u_{t-N} & \cdots & u_{t-1} \end{bmatrix}^T$, $u_f = \begin{bmatrix} u_t & \cdots & u_{t+N_u-1} \end{bmatrix}$ and the dimensions of subspace matrices change to $L_w = (mN_2 \times (l+m)N)$ and $L_u = (mN_2 \times lN_u)$.

The predictor equation in equation (4.12) is used in minimizing the objective function

$$J = \sum_{k=1}^{N_2} (\hat{y}_{t+k|t} - r_{t+k})^2 + \sum_{k=1}^{N_u} \lambda (u_{t+k-1})^2 \qquad (4.13)$$

to derive the 'subspace predictive control' law presented in [23], which computes the future control moves as

$$u_f = (\lambda I + L_u^T L_u)^{-1} L_u^T (r_f - L_w w_p) \qquad (4.14)$$

where $w_p = W_p(:, 1)$. For a finite $\{N_2, N_u\}$, the above control law is called SPC or subspace predictive controller in [23]. As $\{N_2, N_u\} \longrightarrow \infty$, the above control law becomes an LQG-controller presented in [25, 26]. However, for implementation on real processes the controller should have an integrator since the objective function (4.13) does not admit zero static error in the case of non-zero constant reference unless the open loop process contains an integrator [7]. Hence we need to use the GPC objective function, with incremental inputs Δu_f, shown in equation (4.8). One of the several subspace matrices based predictive controller design approaches presented in [103, 104, 106] has also included an integrator. In their method, to get an integrator in the predictor, equation (4.12) is multiplied on both sides with a difference operator, $\Delta = 1 - z^{-1}$, where z^{-1} is the backshift operator, and then rearranged to get a predictor equation with incremental inputs and outputs. A slightly different subspace identification method called DSR [102, 105] is used in their approaches.

In the next section we present a different approach to get incremental variables in the predictor equation. The new approach uses an integrated noise model. As the subspace model in equation (2.19) is I/O based, it is logical to use a similar technique to that adopted in conventional GPC [12, 13]. As will become clear later on, the new approach is equivalent to the original GPC design since the innovations

form state space representation in equations (2.1)-(2.2) combined with integrated noise assumption is equivalent to ARIMAX representation.

4.3.1 Inclusion of integral action through integrated noise model

Consider the noise input e_t as an integrating noise, which is common in the process industries. Therefore,

$$e_{k+1} = e_k + a_k \tag{4.15}$$

$$e_k = \frac{a_k}{\Delta} \tag{4.16}$$

where a_k is a white noise signal and $\Delta\ (= 1 - z^{-1})$ is a differentiating operator. Note that the system considered in equations (2.1)-(2.2) together with (2.10) is equivalent to an ARIMAX representation, as in equation (4.1), considered in the GPC design. Substituting equation (4.15) in (2.1)-(2.2), we obtain

$$z_{k+1} = Az_k + B\Delta u_k + Ka_k \tag{4.17}$$

$$\Delta y_k = Cz_k + D\Delta u_k + a_k \tag{4.18}$$

where $z_k = x_k - x_{k-1}$. The subspace matrix input-output expression for the system (4.17)-(4.18) is now

$$\Delta Y_f = \Gamma_N Z_f + H_N \Delta U_f + H_N^s A_f \tag{4.19}$$

and

$$\Delta \hat{y}_f = \Gamma_N z_k + H_N \Delta u_f \tag{4.20}$$

$$= L_w \begin{bmatrix} \Delta y_p \\ \Delta u_p \end{bmatrix} + L_u \Delta u_f \tag{4.21}$$

Using the system representation (4.17)-(4.18) we can write a k-step ahead predictor as

$$\begin{aligned} y_{t+k-1} - y_{t-1} = {} & (CA^{k-1} + ... + CA + C)z_t + [(CA^{k-2}B + ... + D)\Delta u_t \\ & + ... + D\Delta u_{t+k-1}] + [a_t + a_{t+1} + ... + a_{t+k-1}] \end{aligned} \tag{4.22}$$

and equations (4.9)-(4.10) change to

$$\hat{y}_f = \begin{bmatrix} \hat{y}_{t+1} & \hat{y}_{t+2} & \dots & \hat{y}_{t+N} \end{bmatrix}^T \tag{4.23}$$

$$= \mathbf{y_t} + \Gamma^{\circ}_{N_2} z_t + S_{N_2,N_u} \triangle u_f \tag{4.24}$$

$$= \mathbf{y_t} + L^{\circ}_w(1 : N_2 m, :) \begin{bmatrix} \triangle y_p \\ \triangle u_p \end{bmatrix} + S_{N_2,N_u} \triangle u_f \tag{4.25}$$

$$= F + S_{N_2,N_u} \triangle u_f \tag{4.26}$$

where $\Gamma^{\circ}_{N_2}$ is the modified extended observability matrix and S_{N_2,N_u} is the $(N_2 m \times N_u l)$ dynamic matrix containing the step response coefficients / Markov parameters and formed from L_u.

$$\mathbf{y_t} = \begin{bmatrix} y_t & y_t & \dots & y_t \end{bmatrix}^T \tag{4.27}$$

$$\Gamma^{\circ}_{N_2} = \begin{bmatrix} C \\ CA + C \\ \dots \\ CA^{N_2-1} + \dots + C \end{bmatrix} \tag{4.28}$$

$$S_{N_2,N_u} = \begin{bmatrix} D & 0 & 0 & .. & 0 \\ CB + D & D & 0 & .. & 0 \\ CAB + CB + D & CB + D & D & .. & 0 \\ .. & .. & .. & .. & .. \\ CA^{N_2-2}B + .. + CB + D & CA^{N_2-3}B + .. + CB + D & .. & .. & .. \end{bmatrix}$$

$$= L_u(1 : N_2 m, 1 : N_u l) \begin{bmatrix} I_l & 0 & \dots & 0 \\ I_l & I_l & \dots & 0 \\ \dots & \dots & \dots & \dots \\ I_l & I_l & \dots & I_l \end{bmatrix} \tag{4.29}$$

L°_w is constructed from L_w as

$$L^{\circ}_w(m(k-1)+1 : mk, :) = \sum_{i=1}^{k} L_w(m(i-1)+1 : mi, :) \; 1 \le k \le N_2 \tag{4.30}$$

and F is the free response of the process output.

$$F = \mathbf{y_t} + L^{\circ}_w(1 : N_2 m, :) \begin{bmatrix} \triangle y_p \\ \triangle u_p \end{bmatrix} \tag{4.31}$$

Note that the matrices L_w° and S_{N_2,N_u} are related in a simple manner to L_w and L_u. Even though L_w° and S_{N_2,N_u} can be alternatively directly identified from the differentiated data, it is difficult to design an input signal for such an identification. Hence, a simple strategy is to identify L_w and L_u and use these matrices to form L_w° and S_{N_2,N_u}.

The objective function in equation (4.8) can be expanded as

$$J = (r_f - F - S_{N_2,N_u}\triangle u_f)^T(r_f - F - S_{N_2,N_u}\triangle u_f) + \triangle u_f^T(\lambda I)\triangle u_f \quad (4.32)$$

Differentiating J with respect to $\triangle u_f$ and equating it to zero gives the control law

$$\triangle u_f = (S_{N_2,N_u}^T S_{N_2,N_u} + \lambda I)^{-1} S_{N_2,N_u}^T (r_f - F) \quad (4.33)$$

Only $\triangle u_f(1)$ is implemented and the calculation is repeated at each time instant. Hence at time instant t, we only calculate

$$\triangle u_t = \triangle u_f(1) = m_l \ (r_f - F) \quad (4.34)$$

where m_l is made of the first l-rows of the matrix $(S_{N_2,N_u}^T S_{N_2,N_u} + \lambda I)^{-1} S_{N_2,N_u}^T$. Therefore u_t is implemented as

$$u_t = u_{t-1} + \triangle u_t = u_{t-1} + m_l \ \{r_f - \{\mathbf{y_t} + L_w^\circ(1 : N_2 m, :) \begin{bmatrix} \triangle y_p \\ \triangle u_p \end{bmatrix}\}\} \quad (4.35)$$

Note that the above control law has a guaranteed integral control action and obtained directly from the subspace matrices, without any intermediate parametric model identification step.

Remarks:

Each block-column of the subspace matrix L_u contains the series of process Markov parameters incrementally ordered. Due to the way the variables are arranged in the data Hankel matrices, each block-column of the subspace matrices are identified independently, i.e., each series of Markov parameters in the subspace matrices is identified independently. Hence due to the inherent nature of the subspace approach multiple-models for the process are captured in the subspace matrices to be used for

designing predictive controller. This is clearly an advantage compared to the MPC design via parametric model identification.

4.3.2 Inclusion of feedforward control

If some of the process disturbances are measurable, then with the understanding that measured disturbances are those process input variables which cannot be manipulated for controlling the process outputs, the state space representation of the process (2.1)-(2.2) can be modified as

$$x_{k+1} = Ax_k + \begin{bmatrix} B & B_v \end{bmatrix} \begin{bmatrix} u_k \\ v_k \end{bmatrix} + Ke_k \tag{4.36}$$

$$y_k = Cx_k + \begin{bmatrix} D & D_v \end{bmatrix} \begin{bmatrix} u_k \\ v_k \end{bmatrix} + e_k \tag{4.37}$$

where $v_t(h \times 1)$ is the vector of measured disturbance variables. The matrix input-output equations (2.12) and (2.19) change to

$$Y_f = \Gamma_N X_f^b + H_N U_f + H_N^v V_f + H_N^s E_f \tag{4.38}$$

$$\hat{Y}_f = \Gamma_N X_f^b + H_N U_f + H_N^v V_f \tag{4.39}$$

$$= L_w^b W_p^b + L_u U_f + L_v V_f \tag{4.40}$$

where

$$H_N^v = \begin{bmatrix} D^v & 0 & \dots & 0 \\ CB^v & D^v & \dots & 0 \\ \dots & \dots & \dots & \dots \\ CA^{N-2}B^v & CA^{N-3}B^v & \dots & D^v \end{bmatrix} \tag{4.41}$$

$$W_p^b = \begin{bmatrix} Y_p \\ U_p \\ V_p \end{bmatrix} \tag{4.42}$$

with V_p and V_f being the *past* and *future* data Hankel matrices of v_t (see equations (2.4)-(2.7) for reference). The subspace matrices L_w^b, L_u and L_v are obtained by

finding the prediction of future outputs, Y_f, by solving the least squares problem.

$$\min_{L_w^b, L_u, L_v} \|Y_f - \begin{pmatrix} L_w^b & L_u & L_v \end{pmatrix} \begin{pmatrix} W_p^b \\ U_f \\ V_f \end{pmatrix} \|_F^2 \tag{4.43}$$

The solution is obtained as explained in chapter 2. For predictive control we have the values of measured disturbance only upto the current sampling instant, t, (and do not have the knowledge of the future values of measured disturbance) i.e., v_k for $k = t, t-1, t-2, ...$ are known but v_{t+k} for $k = 1, 2, ..., N_2$ are not available for the prediction of $\hat{y}_{t+k}$. Therefore we can write the prediction expression for $\hat{y}_f$ as

$$\hat{y}_f = \mathbf{y_t} + L_w^{\pm}(1 : N_2 m, :) \begin{bmatrix} \Delta y_p \\ \Delta u_p \\ \Delta v_p \end{bmatrix} + S_{N_2,N_u} \Delta u_f \tag{4.44}$$

$$= F^b + S_{N_2,N_u} \Delta u_f \tag{4.45}$$

where $L_w^{\pm}$ is constructed from L_w^b, and F^b is the free response for the case of measured disturbances.

$$L_w^{\pm}(m(k-1)+1 : mk, :) = \sum_{i=1}^{k} L_w^b(m(k-1)+1 : mk, :) \; 1 \le k \le N_2 \tag{4.46}$$

$$F^b = \mathbf{y_t} + L_w^{\pm}(1 : N_2 m, :) \begin{bmatrix} \Delta y_p \\ \Delta u_p \\ \Delta v_p \end{bmatrix} \tag{4.47}$$

S_{N_2,N_u} is the same as defined before in equation (4.29). Therefore, the feedback plus feedforward control law becomes

$$\Delta u_f = (S_{N_2,N_u}^T S_{N_2,N_u} + \lambda I)^{-1} S_{N_2,N_u}^T (r_f - F^b) \tag{4.48}$$

4.3.3 Constraint handling

Constraints arise due to physical limitations, quality specifications, safety concerns and limiting the wear of the equipment. One of the main features of MPC, its prediction capability, is useful in anticipating constraint violations and correcting

them in an appropriate way [10]. The explicit handling of constraints may allow the process to operate closer to optimal operating conditions [10]. For the constrained case, the computations are more involved. The problem takes the form of a standard Quadratic Programming (QP) formulation and the optimization is done numerically. The quadratic program solved at every instant is

$$\begin{aligned}
\min_{\Delta u} J &= (r_f - F)^T(r_f - F) + \Delta u^T(S_{N_2,N_u}^T S_{N_2,N_u})\Delta u - 2(r_f - F)^T S_{N_2,N_u}\Delta u \\
&= \Delta u^T(S_{N_2,N_u}^T S_{N_2,N_u} + \lambda I)\Delta u - 2(r_f - F)^T S_{N_2,N_u}\Delta u \\
&= \frac{1}{2}\Delta u^T \mathcal{P}\Delta u + \mathcal{C}^T\Delta u \qquad (4.49) \\
s.t. \quad & \mathcal{A}\Delta u \leq \mathcal{B} \qquad (4.50)
\end{aligned}$$

where

$$\mathcal{P} = S_{N_2,N_u}^T S_{N_2,N_u} + \lambda I \qquad (4.51)$$

$$\mathcal{C} = -2S_{N_2,N_u}^T(r_f - F) \qquad (4.52)$$

matrices $\mathcal{A}$ and $\mathcal{B}$ are formed from the constraints (see appendix B). The optimization of the above QP formulation is carried out by means of the standard commercial optimization QP code at each sampling instant and then the value of u_t is sent to the process. Though computationally more involved than the other simpler algorithms, the flexible constraints handling capabilities of predictive controllers are very attractive for practical applications, since the economic operating point of a typical process unit often lies at the intersection of constraints [97, 10]. For more discussion on other types of constraints, for example soft constraints or specifications on the process response characteristics, readers are referred to [10, 29, 78, 99] and the references therein.

4.4 Tuning the noise model

The disturbance dynamics of industrial processes frequently change with time. Tuning of the noise model is a key feature of predictive controller formulations like GPC [7]. It is necessary to incorporate such a feature in the proposed predictive

controller derived from subspace matrices. For this reason we need to separate the state space model of the system into two parts, a deterministic part and a stochastic part which are similar to the process model and noise model in an equivalent input-output transfer function framework as

$$\begin{aligned} y_t &= y_t^d + y_t^s \\ &= [C(zI-A)^{-1}B + D]u_t + [C(zI-A)^{-1}K + 1]e_t \end{aligned} \tag{4.53}$$

It is observed that both deterministic and stochastic parts have the same poles. Hence an equivalent representation for the above equation in the discrete transfer function domain would be an ARMAX model

$$y_t = \frac{G(z^{-1})}{H(z^{-1})}u_t + \frac{F(z^{-1})}{H(z^{-1})}e_t \tag{4.54}$$

$H(z^{-1})$ and $F(z^{-1})$ are monic polynomials in z^{-1}. We can write

$$[C(zI-A)^{-1}K + 1] = 1 + CKz^{-1} + CAKz^{-2} + ... \tag{4.55}$$

The assumption that the noise model $\frac{F(z^{-1})}{A(z^{-1})} = 1$ is equivalent to assuming that the Kalman gain matrix, K, is equal to zero. Therefore if the user desires to change the stochastic part of the identified innovation model, without changing the deterministic model and hence without changing the system matrices C and A, the only way to do it is by changing the Kalman gain matrix, K.

Suppose that the new Kalman gain matrix is represented as K^*. K^* is a matrix $(m \times n)$ for a multiple-output system, and a vector $(1 \times n)$ for a single-output system. We can express K^* as

$$K^* = K + K' \tag{4.56}$$

We can then write

$$\begin{aligned} & [C(zI-A)^{-1}K^* + 1] \\ &= 1 + CK^*z^{-1} + CAK^*q^{-2} + ... && (4.57) \\ &= 1 + C(K+K')z^{-1} + CA(K+K')q^{-2} + ... && (4.58) \\ &= [C(zI-A)^{-1}K + 1] + [CK'z^{-1} + CAK'z^{-2} + ...] && (4.59) \end{aligned}$$

The stochastic part of the output with the new Kalman gain matrix is

$$
\begin{aligned}
(y_t^s)^* &= [C(zI-A)^{-1}K^* + 1]e_t && (4.60)\\
&= [C(zI-A)^{-1}K + 1]e_t + [CK'z^{-1} + CAK'z^{-2} + ...]e_t && (4.61)\\
&= y_t^s + [CK'z^{-1} + CAK'z^{-2} + ...]e_t && (4.62)\\
&= y_t^s + \begin{bmatrix} C & CA & ... & CA^{n-1} \end{bmatrix} K' \begin{bmatrix} e_{t-1} \\ e_{t-2} \\ ... \\ e_{t-n} \end{bmatrix} && (4.63)\\
&= y_t^s + (\Gamma_n K')^T e_p && (4.64)
\end{aligned}
$$

where $e_p = \begin{bmatrix} e_{t-1} & e_{t-2} & ... & e_{t-n} \end{bmatrix}^T$ and Γ_n is the observability matrix which can be estimated implicitly in the subspace identification method without having to first calculate the system matrices C and A, through the SVD approach by inspecting the number of dominant singular values in the singular value decomposition of $L_w W_p$.

$$
\begin{aligned}
L_w W_p &= \begin{pmatrix} U_1 & U_2 \end{pmatrix} \begin{pmatrix} S_1 & 0 \\ 0 & S_2 \end{pmatrix} \begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix} \approx U_1 S_1 V_1^T \quad as \quad j \longrightarrow \infty && (4.65)\\
\Gamma_n &= U_1 S_1^{1/2} && (4.66)
\end{aligned}
$$

If n is the number of dominant singular values taken in S_1, then Γ_N will be an $(mN \times n)$ matrix, where N is the number of block rows taken in W_p. Since the knowledge of the state space system matrices A and C is not required, the new stochastic model can be incorporated in a model free manner. Now the prediction with the "tuned" noise model can be written as

$$
(\hat{y}_t)^* = \hat{y}_t + (\Gamma_n K')^T \; e_p = \hat{y}_t + \gamma_n \; e_p \qquad (4.67)
$$

where $\gamma_n = (\Gamma_n K')^T$, which can be considered as a vector of impulse response coefficients (Markov parameters for the multivariate systems) with the new noise model. γ_n is constructed from the estimated observability matrix, Γ_n, and the user specified $(n \times m)$ matrix, K'. Noise model tuning is used as a tool to make up for the process-model mismatch resulting from changes of the process from time to time or simply as tuning parameters. e_p contains the past prediction errors and can

be estimated from the data as one step ahead prediction errors. In essence adding the term $[(\Gamma_n K')^T e_p]$ is equivalent to filtering the past prediction errors. Hence K' is used as a tuning parameter and is chosen in such a way that it minimizes the prediction errors.

Thus incorporating a new noise model simply involves the addition of a new term in the calculation of the free response of the process. Hence the free response calculation, equation (4.31), modifies as

$$F = \mathbf{y_t} + L_w^{\circ}(1 : N_2 m, :) \begin{bmatrix} \Delta y_p \\ \Delta u_p \end{bmatrix} + \Upsilon e_p \tag{4.68}$$

where Υ $(N_2 m \times n)$ is a left-upper triangular matrix constructed from the elements of γ_n.

$$\Upsilon = \begin{bmatrix} \gamma_n(1) & \gamma_n(2) & \dots & \gamma_n(n-1) & \gamma_n(n) \\ \gamma_n(2) & \gamma_n(3) & \dots & \gamma_n(n) & 0 \\ \dots & \dots & \dots & \dots & \dots \\ \gamma_n(N_2) & \dots & \gamma_n(n) & \dots & 0 \end{bmatrix} \tag{4.69}$$

4.5 Simulations

The proposed control design method is tested in simulations. The system example is taken from MATLAB/MPC toolbox working example.

$$\begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix} = \begin{bmatrix} \frac{12.8e^{-s}}{16.7s+1} & \frac{-18.9e^{-3s}}{21.0s+1} \\ \frac{6.6e^{-7s}}{10.9s+1} & \frac{-19.4e^{-3s}}{14.4s+1} \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix} + \begin{bmatrix} \frac{3.8e^{-8s}}{14.9s+1} \\ \frac{4.9e^{-3s}}{13.2s+1} \end{bmatrix} w(s) + \begin{bmatrix} e_1(s) \\ e_2(s) \end{bmatrix} \tag{4.70}$$

Open loop input/output data is obtained by exciting the open loop system using a designed 'RBS' signal of magnitude 1 for the inputs, $\mathbf{u_k}$ and random numbers of standard deviation 0.1 for the white noise sequences, $\mathbf{e_k}$, in MATLAB-*Simulink*. A random walk signal is designed for the measured disturbance w_k by passing a white noise signal of standard deviation 0.1 through an integrator. Sampling interval is

taken as 2 units of time. Using subspace identification, with $N = 50$ (row blocks) and $j = 2000$ (column blocks) in the data Hankel matrices, the subspace matrices $L_w(100 \times 250)$, $L_u(100 \times 100)$ and $L_v(100 \times 50)$ are identified. The simulation data and the models from subspace matrices are plotted in (4.1)-(4.2). As can be seen from (4.2), the impulse response models from the identified subspace matrices match very well with the true impulse response models. Note that even though the signal used for the measured disturbance is not a white noise signal, we can still identify the model corresponding to the measured disturbance very accurately.

In figure (4.3) the simulation results with subspace based predictive controller without an integrator (SPC in [23]) is compared with the predictive controller with integral action. As illustrated, the controller with integrator gives no offset for non-zero setpoints. In figure (4.4) the predictive controller performance is compared for the cases without and with feedforward control. Better controller performance has been achieved with feedforward control.

For a range of values for λ, N_2, N_u and constraints on the input moves, $\Delta\mathbf{u}$, a subspace matrices based predictive controller is implemented on the above process in MATLAB-*Simulink*. The closed loop system response for different sets of tuning parameters is illustrated in figures (4.5)-(4.8).

In figure (4.5) it can be seen that as the weighting, λ, on the input increases the controller response becomes less aggressive. For a given prediction horizon, as the control horizon, N_u, increases, the controller gives more aggressive tracking performance as shown in figure (4.6). For a given control horizon, as the prediction horizon, N_2, increases the controller gives better setpoint tracking performance (4.7). Figure (4.8) shows the setpoint tracking under different constraints on the incremental control moves, $\Delta\mathbf{u}$. It can be seen that smaller the magnitude of the maximum allowed control moves, more sluggish is the controller response to setpoint changes.

Noise model tuning

To illustrate the tuning of noise model with the subspace approach consider the

process model changes with time, in other words there is a mismatch between the true process model and the identified process model used in the controller design. Consider the case when the process model from equation (4.70) changes to

$$\begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix} = \begin{bmatrix} \frac{12.8e^{-s}}{16.7s+1} & \frac{-18.0e^{-3s}}{21.0s+1} \\ \frac{8e^{-7s}}{10.9s+1} & \frac{-18e^{-3s}}{14.4s+1} \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix} + \begin{bmatrix} \frac{3.8e^{-8s}}{14.9s+1} \\ \frac{4.9e^{-3s}}{13.2s+1} \end{bmatrix} w(s) + \begin{bmatrix} e_1(s) \\ e_2(s) \end{bmatrix} \quad (4.71)$$

Figure (4.9) illustrates the controller response without and with the on-line noise model tuning feature.

4.6 Experiment on a pilot scale process

The proposed predictive controller is tested on a multivariate pilot scale system. The system considered, shown in figure (4.10), is a three tank system with two inlet water flows. The levels of Tank-1 and Tank-2 are the two controlled variables (CVs). The setpoints (SPs) for the flow rates through the valves-A & B are the two manipulated variables (MVs). The flow rates through the valves-A & B are controlled through the *local*-PID controllers on each valve. The setpoints for the flow rates come from a higher level advanced controller application. The *local*-PID controllers, which are univariate, are at faster sampling (1 sec). The higher level controller, which is multivariate and does computations to minimize an optimization function, sends controller outputs every 6 seconds. The system is configured so as to emulate a typical multivariate system in the industries.

Tank-3 and valve-C are used primarily to introduce interactions between the variables in the system. As can be seen in figure (4.10) a change in the level in tank-1 effects the level in tank-2 *via* tank-3 level. The degree of interaction can be manipulated by changing the valve-C position. If the valve-C is completely closed then the level in tank-2 is independent of the level in tank-1 (zero interactions). By opening the valve-C interactions are introduced in the tank-2 level. Valve-C is

maintained at a fixed open position throughout the exercise. Note that the level in tank-1 is independent of the levels in tank-2 & 3. The step response models for the system, which are formed from the impulse response coefficients in the subspace matrices, are plotted in figure (4.11). The correlations between the variables are clear from the step response plots.

Open loop step-test data for the system is collected by sending two (uncorrelated) designed 'PRBS' signals for the SPs of the flow rates through valves-A & B. Subspace matrices are identified using the open loop data. A multivariate subspace matrices based predictive controller is then designed for the system. The controller parameters (weighting matrices, prediction horizon, control horizon and noise model) are tuned for a smooth controller performance. The closed loop response for the unconstrained and constrained ($|\Delta u| \leq 0.5$) cases are plotted in figure (4.12) and figure (4.13) respectively.

4.7 Conclusions

In this chapter, the design of the predictive controller, in the GPC framework, using the subspace matrices calculated through the subspace identification method is addressed. Important issues in practical implementation of the predictive controllers such as integral action, constraint handling and feedforward control are discussed. It has been shown that the noise model can be independently specified by the user through the addition of a new term to the predictor equation in the model-free manner, which is shown to be equivalent to changing the Kalman filter gain matrix. The equivalence of the predictive controller designed from subspace matrices to the traditional GPC is shown in appendix D. The proposed predictive controller is tested on multivariate systems in simulations and on a pilot scale process.

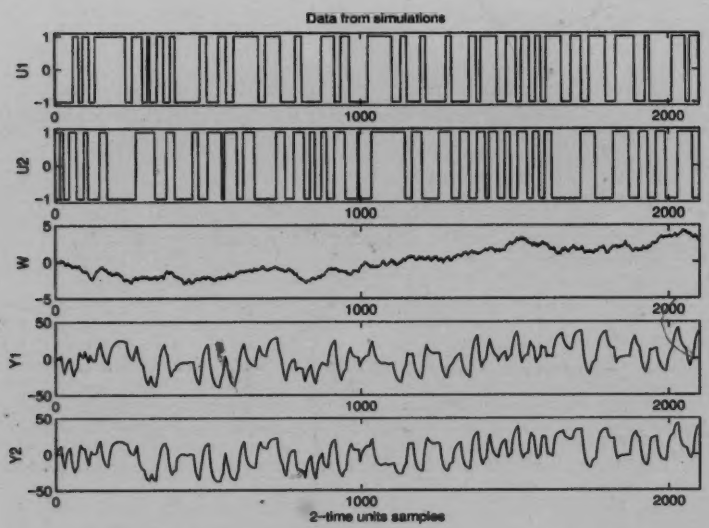


Figure 4.1: Inputs, measured disturbance and outputs data from simulations.

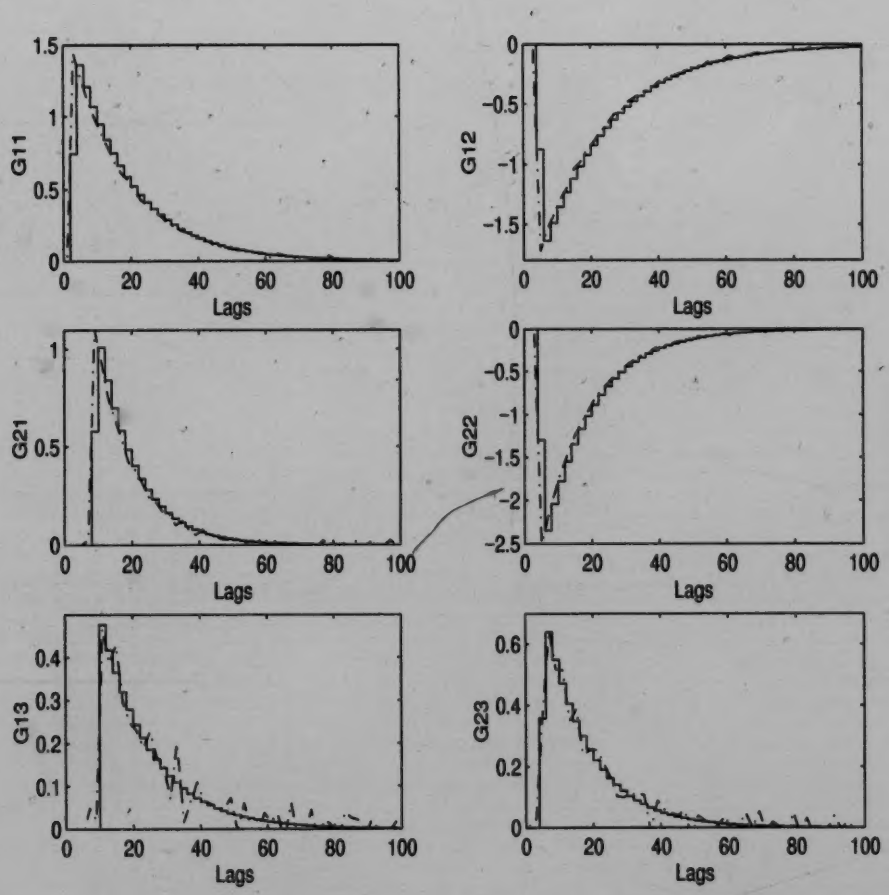


Figure 4.2: Comparison of the process and noise models from subspace matrices with the true models.

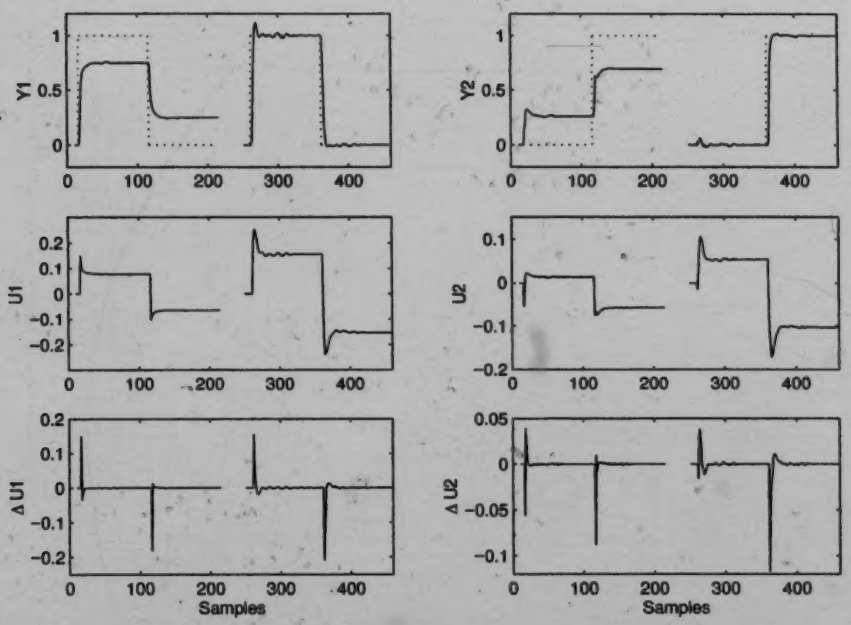


Figure 4.3: Predictive controller without and with the integrator. w(t)=0; e(s) =0.

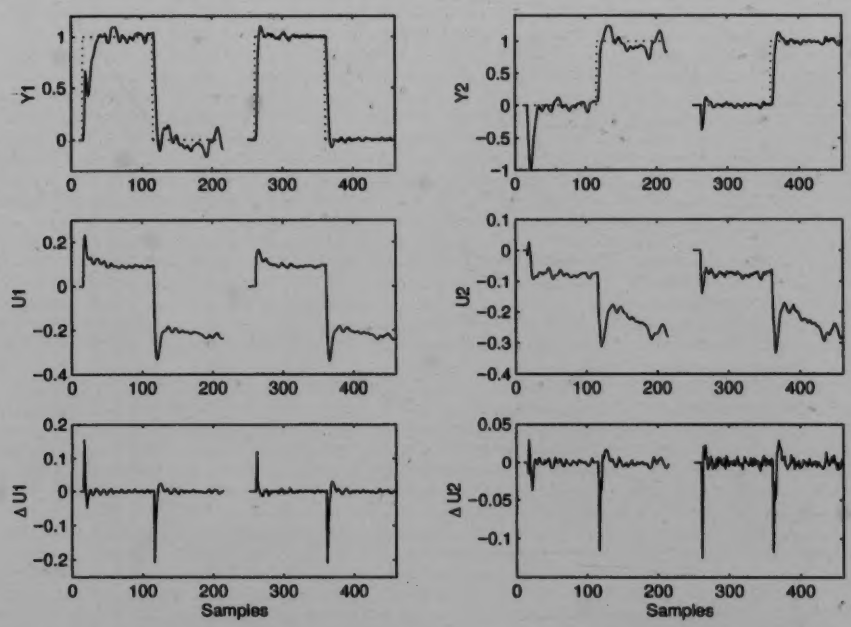


Figure 4.4: Predictive controller without and with the feedforward control. e(s) =0.

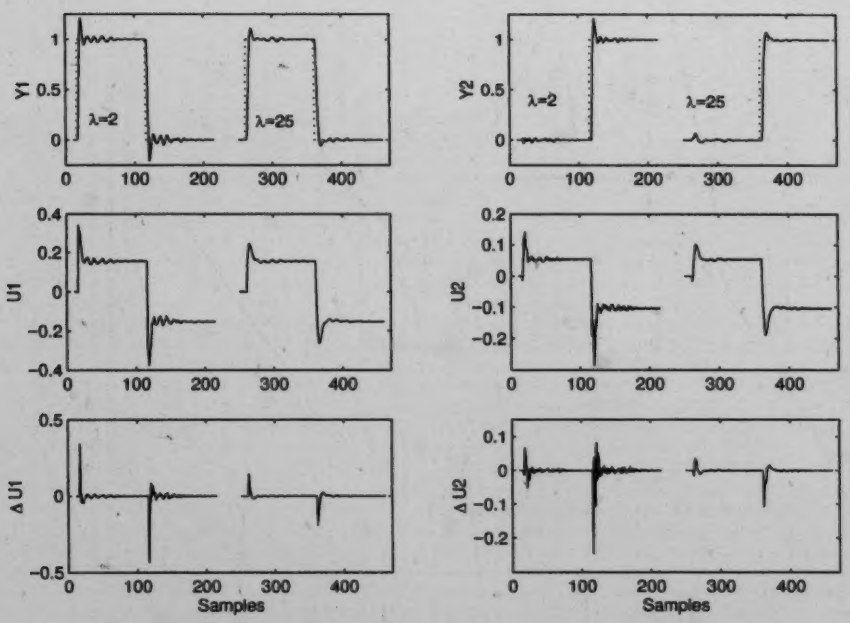


Figure 4.5: Variation of input weighting, λ. w(t)=0; e(s) =0.

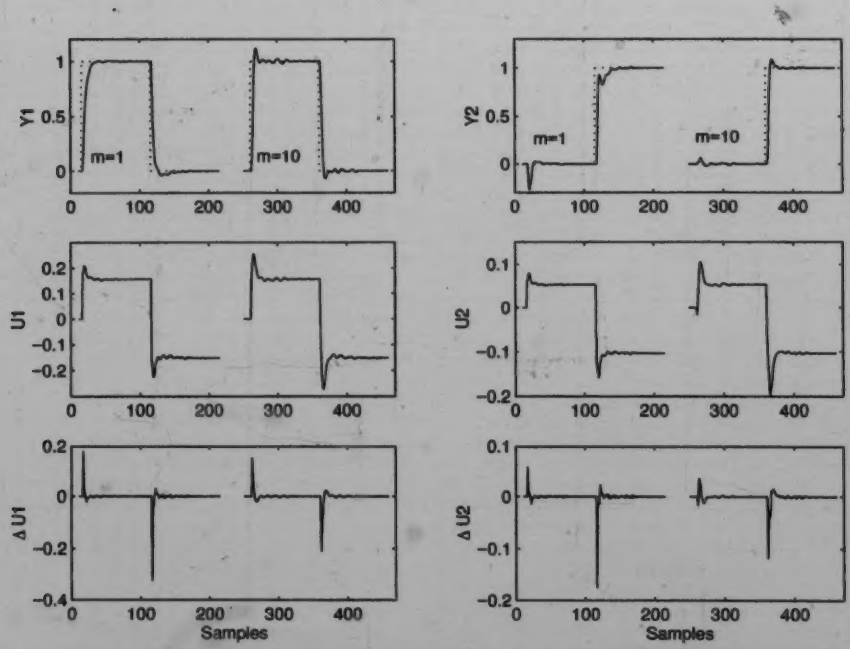


Figure 4.6: Variation of control horizon, N_u. w(t)=0; e(s) =0.

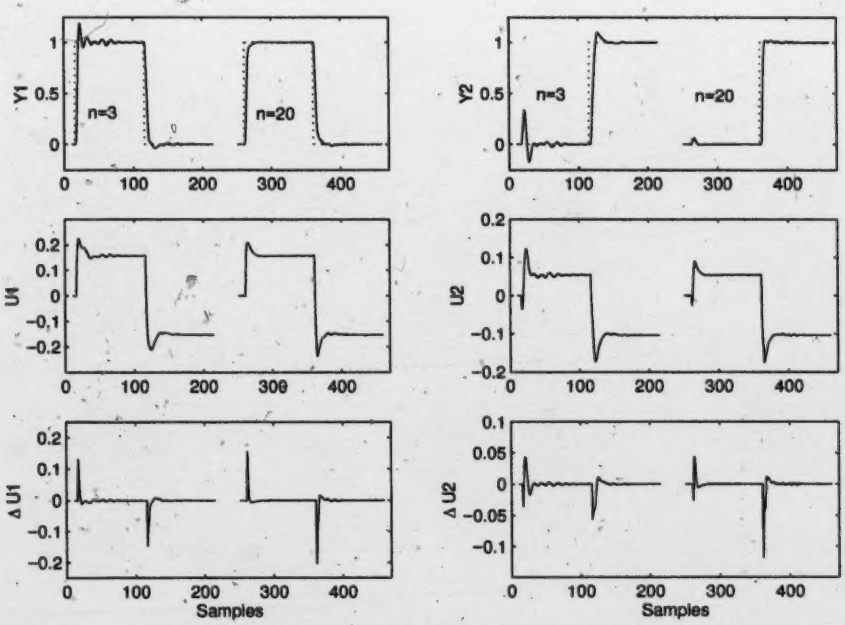


Figure 4.7: Variation of prediction horizon, N_2. w(t)=0; e(s) =0.

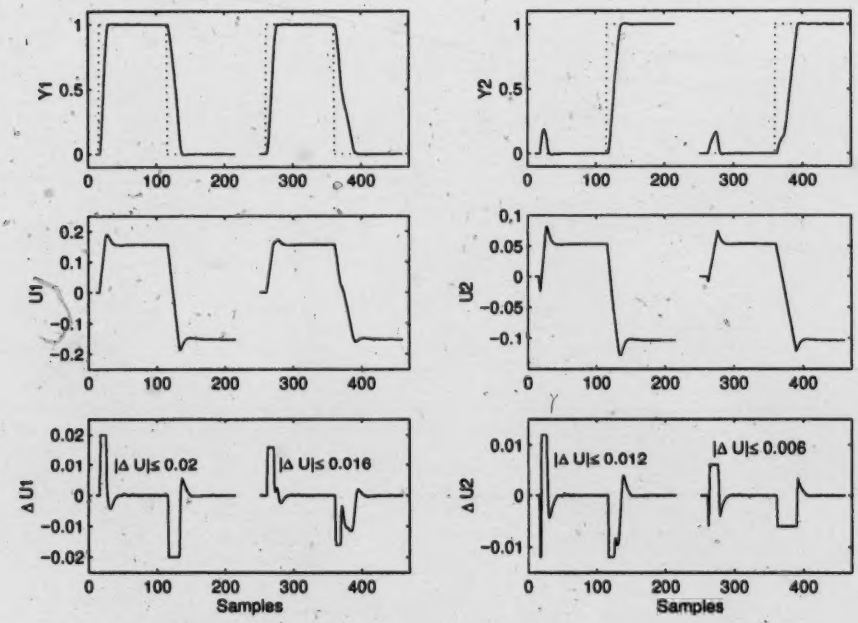


Figure 4.8: Constrained predictive controller. w(t)=0; e(s) =0.

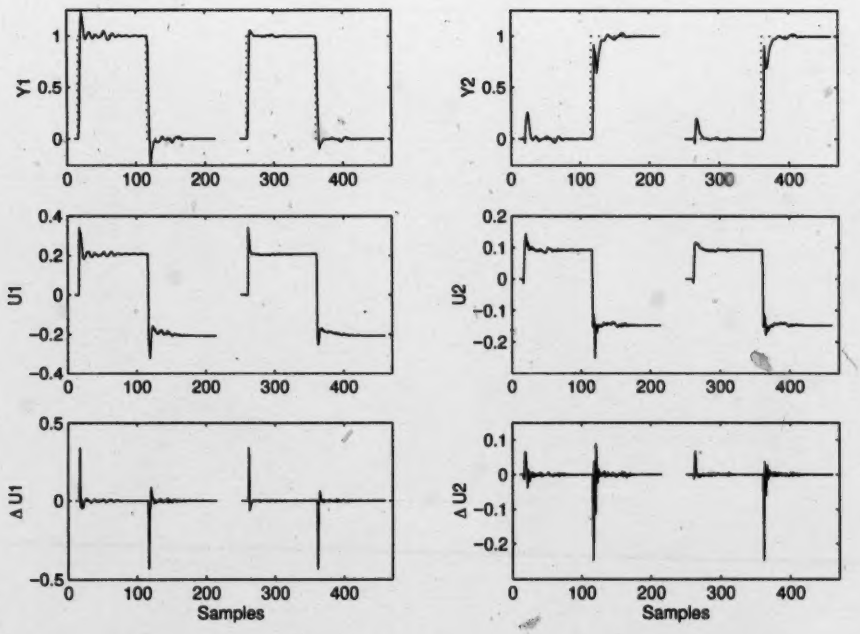


Figure 4.9: Noise model tuning for model mismatch. w(t)=0; e(s) =0.

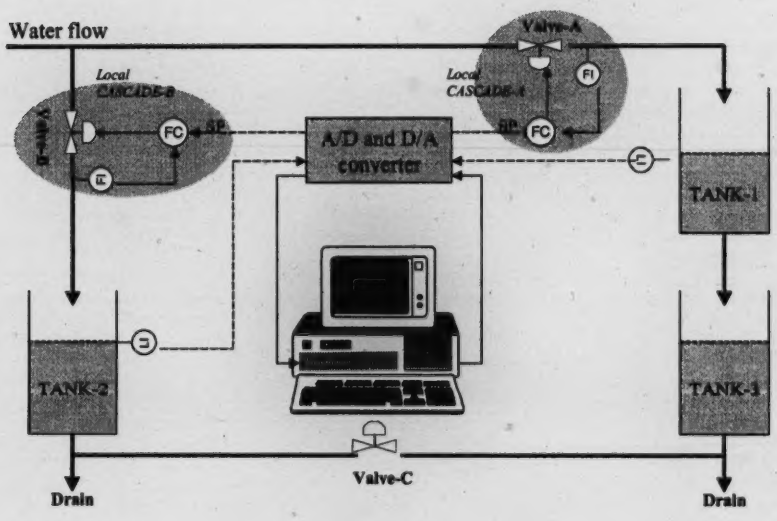


Figure 4.10: Experimental setup

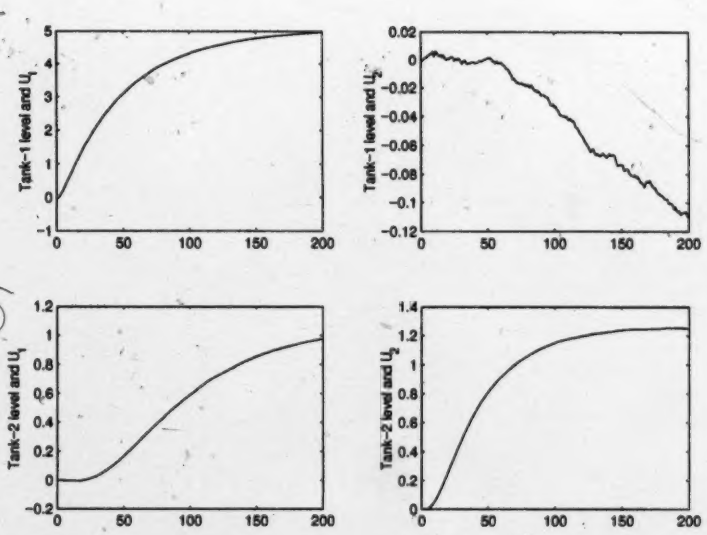


Figure 4.11: Step response coefficients from the subspace matrices identified using the open loop data.

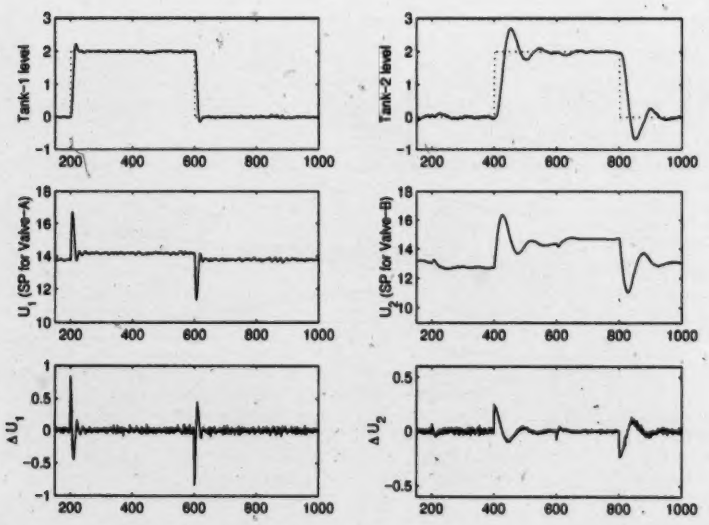


Figure 4.12: Subspace based predictive controller on the pilot scale process.

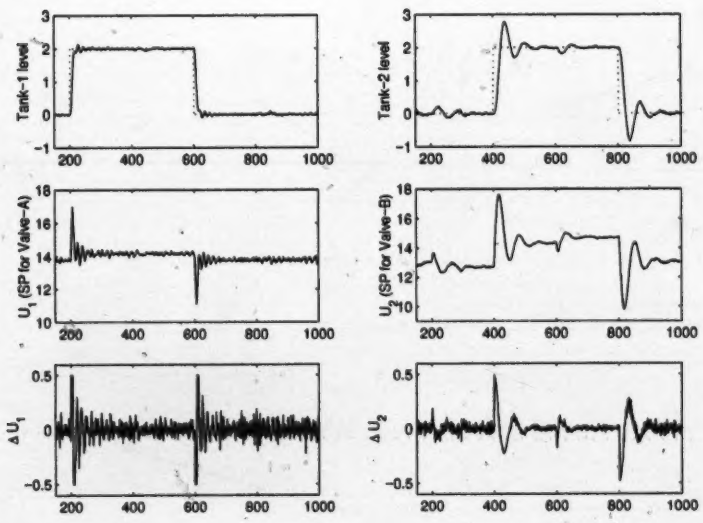


Figure 4.13: Subspace based predictive controller for the constrained case on the pilot scale process.

Chapter 5

Controller performance analysis with LQG-benchmark obtained under closed loop conditions

[1]

5.1 Introduction

A typical industrial plant can contain thousands of controllers ranging from PI/PID controllers to the more advanced model predictive controllers like dynamic matrix control (DMC) [15, 16], Generalized predictive controller (GPC) [12, 13], Quadratic dynamic matrix controller (QDMC) [30], etc. With a goal towards optimal performance, energy conservation and cost effectiveness of the process operations in the industry, controller performance assessment has been receiving attention both from the industry and from the academia since the notable work of Harris [34]. Periodic tuning of the controllers becomes an important task of control

[1] A version of this chapter has been accepted for publication as a journal paper
R. Kadali, and B. Huang. Controller performance analysis with LQG benchmark obtained under closed loop conditions. *ISA Transactions*, 41; 2002.

engineers for obtaining optimal performance from the control systems. Controller performance assessment techniques are used as a tool to check the optimality of the current controller tuning parameters settings. Several benchmarks such as minimum variance control (MVC) [5, 19, 20, 34, 35, 36, 38, 42, 45, 46, 50, 82, 83, 86, 98, 111], linear quadratic Gaussian (LQG) control [42, 46] and designed controller performance versus achieved controller performance [69, 70, 96], etc. have been proposed for assessing the controller performance. Among these approaches, MVC-benchmark is one of the popular benchmarks due to its non-intrusive nature for the univariate systems and routine closed loop operating data can be used for the calculation of this benchmark. For the univariate application, only *a priori* knowledge of the process time delay is required for obtaining the MVC-benchmark from routine operating data [5, 19, 20, 46, 38, 37, 34, 82, 98, 111, 114]. For the multivariate systems, the calculations are more involved and require estimation of the unitary interactor matrix [42, 35, 36, 38, 45, 50, 48, 51, 60, 86, 98, 125]. However the MVC-benchmark may not be a practical one for those control systems whose objective is not just minimizing process output variance but also keeping the input variability (for example, valve movement) within some specified range to reduce upset to other processes, conserve energy and lessen the equipment wear. The objective of such controllers may be expressed as minimizing a linear quadratic function of input and output variances. The LQG-benchmark is a more appropriate benchmark for assessing the performance of such controllers. However, calculation of the LQG-benchmark requires a complete knowledge of the process model [42, 46], which is a demanding requirement or simply not possible in practice. An open loop test for obtaining the process model may not always be feasible or may be expensive. Frequency domain approach is proposed by Kammer ([62, 63, 64]) for testing the LQ optimality for the performance assessment of a controller using closed loop data with setpoint excitation. However this approach does not give the quantitative values for the controller performance in terms of process input and output variances. In other words, it does not separate the non-optimality/optimality with respect to process response (output) variance and process input variance. In this chapter we propose a subspace matrices based approach to obtain the LQG-benchmark variances of the

process input and output to be used for the controller performance assessment. The required subspace matrices, those corresponding to the deterministic and stochastic inputs, are estimated from closed loop data with setpoint excitation. The method proposed is applicable to both univariate and multivariate systems.

Subspace identification methods allow estimation of a state space model for the system directly from the process data. Certain subspace matrices, corresponding to the states, deterministic inputs and stochastic inputs, are identified as an intermediate step in the subspace identification methods. Several approaches, such as N4SID [93] (**N**umerical **s**ubspace **s**tate space **id**entification), MOESP [119, 120, 121] (**MIMO** **o**utput **e**rror **s**tate space model identification) and CVA [72, 73, 74, 75, 76, 77, 107] (**C**anonical **v**ariate **a**nalysis), are popular for subspace identification using open loop data. Subspace identification methods also exist for closed loop data. Recently Van Overschee and De Moor [94] proposed a subspace identification method for the identification of the subspace matrices (all the three matrices, corresponding to the states, deterministic input and stochastic input) of the process using closed loop data with the knowledge of the first N impulse response coefficients (Markov parameters for the multivariate systems) of the controller, where N is the maximum order of the state space model we want to identify. MOESP and CVA approaches were also used for the identification of a state space model using closed loop data [73, 113, 122]. In addition to the setpoint excitation, MOESP/ CVA approach uses an external white noise signal addition to the controller output to make it independent of the noise. The closed-loop state space model is first identified using the closed loop data from which the open loop state space matrices are retrieved.

Ljung and McKelvey [81] presented a method for the identification of subspace matrices from closed loop data using estimated predictors and state that their algorithm is an illustration of a 'feasible' method rather than the 'best way' of identifying systems operating in closed loop. The primary goal of all the above approaches is the identification of a state space model for the open loop system.

Favoreel et al [24, 25, 26] have recently proposed a method for the design of optimal LQG controllers directly from the subspace matrices, instead of using a state space model. Recent work by Kadali and Huang [58] (adapted as chapter 3 in this thesis) allows identification of (only two of the subspace matrices, corresponding to) the deterministic subspace matrix and stochastic subspace matrix from closed loop data without requiring any *a priori* knowledge of the controllers. This method requires setpoint excitation and is also extended to the case of measured disturbances [58] (see chapter 3). It provides the tools/means for the calculation of more practical controller performance benchmarks like LQG-benchmark using closed loop data. As will be shown later in this chapter, the explicit process model is not required for obtaining the LQG-benchmark.

The method for designing the optimal LQG controller directly from subspace matrices proposed in [24, 25, 26] is extended in this chapter to the case of feedforward plus feedback control. If some of the disturbance variables are measurable, analysis of feedforward control performance is a worthwhile study. However, this analysis requires the subspace matrix corresponding to the measured disturbance variables. Using the subspace approach proposed in [58] the subspace matrix corresponding to the measured disturbance variables can also be estimated under closed loop conditions, if the measured disturbances are assumed to be uncorrelated with the setpoint changes. This provides a means for the profit analysis of implementing feedforward control on the process.

The main contributions of this chapter in the order of presentation are: (*i*) derivation of the expressions for the calculation of the optimal LQG-benchmark variances of the process input and output directly from the subspace matrices, (*ii*) extension of the design of optimal LQG controllers using subspace matrices proposed in [24, 25, 26] to the feedforward plus feedback control case, (*iii*) extension of the analysis to the case of feedforward controller performance analysis, and (*iv*) illustration of the proposed method through an application on a pilot scale process.

This chapter is arranged as follows. Section 5.2 explains the design of the LQG

controller directly from the subspace matrices. Section 5.3 is the main section where the methodology of obtaining the LQG-benchmark variance for the process input and output from closed loop data is presented. Incorporation of feedforward control in the optimal LQG control is discussed in section 5.4. Controller performance analysis indices are defined and described in section 5.5. A summary of the proposed method is presented in section 5.6. Simulation results are presented in section 5.7 followed by an application on a pilot scale process in section 5.8. Conclusions are provided in section 5.9.

5.2 Designing LQG-controller using subspace matrices

A linear time-invariant system can be described in a state space innovations form as:

$$x_{k+1} = Ax_k + Bu_k + Ke_k \tag{5.1}$$

$$y_k = Cx_k + Du_k + e_k \tag{5.2}$$

where x_k, y_k, u_k and e_k are the process states, outputs, deterministic inputs and stochastic inputs respectively. K is the Kalman filter gain and e_k is an unknown innovations sequence of white noise with the covariance matrix S. For an l-input and m-output system, A, B, C, D, K and S are $(n \times n)$, $(n \times l)$, $(m \times n)$, $(m \times l)$, $(n \times m)$ and $(m \times m)$ matrices respectively, where n is the state order.

Using the predictor equations in equations (2.12) and (2.19) for the output of system in equations (2.1)-(2.2), we can write

$$y_f = \Gamma_N x_{t+1} + H_N u_f + H_N^s e_f \tag{5.3}$$

$$= L_w w_p + L_u u_f + L_e e_f \tag{5.4}$$

where

$$y_f = \begin{bmatrix} y_{t+1} \\ \dots \\ y_{t+N} \end{bmatrix}; u_f = \begin{bmatrix} u_{t+1} \\ \dots \\ u_{t+N} \end{bmatrix}; w_p = \begin{bmatrix} y_p \\ u_p \end{bmatrix}; y_p = \begin{bmatrix} y_{t-N+1} \\ \dots \\ y_t \end{bmatrix}; u_p = \begin{bmatrix} u_{t-N+1} \\ \dots \\ u_t \end{bmatrix}$$

and Γ_N $(Nm \times n)$ is the extended observability matrix, H_N $(Nm \times Nl)$ and H_N^s $(Nm \times Nm)$ are the lower triangular Toeplitz matrices containing the impulse response coefficients (Markov parameters) corresponding to the deterministic input u_k and the unknown stochastic input e_k respectively. p and f denote the past and the future respectively. The subscript N follows from the number of steps ahead predictions represented in y_f. L_w $(Nm \times N(l+m))$, L_u $(Nm \times Nl)$ and L_e $(Nm \times Nm)$ are the subspace matrices corresponding to the states,the deterministic inputs and the stochastic inputs respectively.

$$\Gamma_N = \begin{bmatrix} C \\ CA \\ \dots \\ CA^{N-1} \end{bmatrix};$$

$$H_N = \begin{bmatrix} D & 0 & \dots & 0 \\ CB & D & \dots & 0 \\ \dots & \dots & \dots & \dots \\ CA^{N-2}B & CA^{N-3}B & \dots & \dots \end{bmatrix}; \quad H_N^s = \begin{bmatrix} I_m & 0 & \dots & 0 \\ CK & I_m & \dots & 0 \\ \dots & \dots & \dots & \dots \\ CA^{N-2}K & CA^{N-3}K & \dots & \dots \end{bmatrix}$$

Recent work by [25, 24, 26] shows the design of the optimal LQG controller by directly using the subspace matrices, instead of through a state space model, for the system in equations (2.1)-(2.2). The linear quadratic Gaussian (LQG) controller is designed to minimize the following quadratic cost function J over the horizon N:

$$J = E\{ \sum_{k=1}^{N} [(y_{t+k} - r_{t+k})^T \ (y_{t+k} - r_{t+k}) \ + u_{t+k}^T \ (\lambda I_l) \ u_{t+k}]\} \quad (5.5)$$

$$= \sum_{k=1}^{N} [(\hat{y}_{t+k} - r_{t+k})^T \ (\hat{y}_{t+k} - r_{t+k}) \ + u_{t+k}^T \ (\lambda I_l) \ u_{t+k}] \quad (5.6)$$

where E is the expectancy operator, λ is the user defined non-negative input weighting parameter and r_t is the reference for output trajectory. I_l is an l-order identity matrix. $\hat{y}_{t+k}$ is the k-step ahead predicted output given the past inputs and outputs and future inputs upto time t.

It should be noted that traditionally the following objective function is used for the

design of the LQG controllers:

$$J = E\{ \sum_{k=1}^{N} [(y_{t+k} - r_{t+k})^T \; R \; (y_{t+k} - r_{t+k}) \; + u_{t+k}^T \; Q \; u_{t+k}] \; \} \qquad (5.7)$$

where R $(m \times m)$ and Q $(l \times l)$ are non-negative definite weighting matrices. To simplify the presentation the objective function in equation (5.5) is used throughout this chapter. Equation (5.5) basically means that all the inputs have the same weighting (or equal importance) in minimizing the objective function.

The optimal predictor equation from equation (5.4) is:

$$\hat{y}_f = L_w w_p + L_u u_f \qquad (5.8)$$

The notation in the cost function can be simplified for regulatory control, by letting $r_{t+k} = 0$, as:

$$J = \min_{u_f^2} \; [\hat{y}_f^T \; \hat{y}_f + u_f^T \; (\lambda I_{Nl}) \; u_f] \qquad (5.9)$$

$$= (L_w w_p + L_u u_f)^T (L_w w_p + L_u u_f) + u_f^T (\lambda I_{Nl}) u_f \qquad (5.10)$$

Partial differentiation of J with respect to u_f and setting it to zero yields the LQG control law [25] as:

$$u_f = -(\lambda I_{Nl} + L_u^T L_u)^{-1} L_u^T L_w w_p \qquad (5.11)$$

The above control law is the optimal LQG control law as $N \longrightarrow \infty$ and is equivalent to an estimated state feedback control law.

$$u_f = -C_{lqg} \; \hat{x}_{t+1} \qquad (5.12)$$

where

$$C_{lqg} = (\lambda I_{Nl} + L_u^T L_u)^{-1} L_u^T \Gamma_N \qquad (5.13)$$

and the relation $L_w w_p = \Gamma_N \hat{x}_{t+1}$ follows from the equations (5.3)-(5.4). Only the first control move is implemented and the calculation is repeated at each sampling interval.

5.3 Obtaining LQG benchmark from closed loop data

To assess the controller performance, we compare the current controller performance, in terms of output (or input or both) variances, with the variances under the optimal control. This gives rise to the question of selection of the optimal benchmark controller. Though the primary objective of a control system is often to minimize the output variance, we may also want to limit input variance for reasons like energy conservation and equipment wear. In other words, a compromise between the process input variance and output variance is necessary. The optimal LQG control is one of such benchmarks that takes into account both input and output variances of the process and represents a limit of performance in terms of input and output variances [8, 42, 46].

To obtain the optimal LQG-benchmark variances we need to obtain the closed loop expressions for process input and output in terms of the disturbances entering the process. From the predictor equation (5.4), we can write y_{t+1} in terms of the past inputs and past outputs as

$$y_{t+1} = l_{y_p} y_p + l_{u_p} u_p + G_0 u_{t+1} + L_0 e_{t+1} \tag{5.14}$$

where

$$l_{y_p} = L_w(1:m, 1:mN) \tag{5.15}$$

$$l_{u_p} = L_w(1:m, mN+1:(l+m)N) \tag{5.16}$$

and the notation $A(i:j, p:q)$ represents the rows i to j and columns p to q of the matrix A. Equation (5.14) can be transformed to alternatively express the process output in terms of past inputs and past noise with the process and noise model impulse response coefficients (see appendix D) as

$$\begin{aligned} y_{t+1} = & \begin{bmatrix} G_1 & \dots & G_N \end{bmatrix} \begin{bmatrix} u_t \\ \dots \\ u_{t-N+1} \end{bmatrix} + \begin{bmatrix} L_1 & \dots & L_N \end{bmatrix} \begin{bmatrix} e_t \\ \dots \\ e_{t-N+2} \end{bmatrix} \\ & + G_0 \; u_{t+1} + L_0 \; e_{t+1} \end{aligned} \tag{5.17}$$

where G_i and L_i are the i-th impulse response coefficients (Markov parameters for multivariate systems) of the process and noise models respectively. In other words we can express the past (state) contribution term, $L_w w_p$, as

$$l_w w_p = \begin{bmatrix} G_1 & \dots & G_N \end{bmatrix} \begin{bmatrix} u_t \\ \dots \\ u_{t-N+1} \end{bmatrix} + \begin{bmatrix} L_1 & \dots & L_N \end{bmatrix} \begin{bmatrix} e_t \\ \dots \\ e_{t-N+2} \end{bmatrix}$$

$$\Longrightarrow \quad L_w w_p = \begin{bmatrix} G_1 & \dots & G_{N-1} & G_N \\ G_2 & \dots & G_N & 0 \\ \dots & \dots & \dots & \dots \\ G_N & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} u_t \\ \dots \\ u_{t-N+1} \end{bmatrix} + \begin{bmatrix} L_1 & \dots & L_{N-1} & L_N \\ L_2 & \dots & L_N & 0 \\ \dots & \dots & \dots & \dots \\ L_N & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} e_t \\ \dots \\ e_{t-N+1} \end{bmatrix} \tag{5.18}$$

However, the controller output, u_{t+1} is calculated using all the data available at time '$t+1$', i.e., $\{u_t, y_{t+1}, u_{t-1}, y_t, \dots\}$. Hence the original subspace predictor expression in equation (5.4) and the subspace based LQG-controller law in equation (5.11) have to be modified to obtain the closed loop expressions for u_f and y_f. First, define

$$K_{lqg} = (\lambda I_{Nl} + L_u^T L_u)^{-1} L_u^T; \tag{5.19}$$

$$L_g = \begin{bmatrix} G_1 & G_2 & \dots & G_{N-1} & G_N \\ G_2 & G_3 & \dots & G_N & 0 \\ \dots & \dots & \dots & \dots & \\ G_N & 0 & 0 & \dots & 0 \end{bmatrix}; \quad \tilde{u}_p = \begin{bmatrix} u_t \\ u_{t-1} \\ \dots \\ u_{t-N+1} \end{bmatrix}; \tag{5.20}$$

$$L_h = \begin{bmatrix} L_0 & L_1 & \dots & L_{N-1} & L_N \\ L_1 & L_2 & \dots & L_N & 0 \\ \dots & \dots & \dots & \dots & \dots \\ L_{N-1} & 0 & 0 & \dots & 0 \end{bmatrix}; \quad \tilde{e}_p = \begin{bmatrix} e_{t+1} \\ e_t \\ \dots \\ e_{t-N+1} \end{bmatrix}; \tag{5.21}$$

$$\tilde{L}_e = \begin{bmatrix} 0 & 0 & \dots & 0 \\ L_0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ L_N & L_{N-1} & \dots & 0 \end{bmatrix}; \quad \tilde{e}_f = \begin{bmatrix} e_{t+2} \\ e_{t+3} \\ \dots \\ e_{t+N+1} \end{bmatrix} \tag{5.22}$$

Note that the matrices L_g and L_h contain the impulse response coefficients (Markov parameters) for the deterministic and stochastic inputs and can be formed using the subspace matrices L_u and L_e respectively. From equation (5.11), we can write

$$u_f = -(\lambda I_{Nl} + L_u^T L_u)^{-1} L_u^T L_w w_p \tag{5.23}$$

$$= -K_{lqg}\{L_g \tilde{u}_p + L_h \tilde{e}_p\} \tag{5.24}$$

Similarly substituting equation (5.24) in equation (5.4) we can write

$$y_f = L_g \tilde{u}_p + L_h \tilde{e}_p + L_u u_f + L_e e_f \tag{5.25}$$

$$= (I - L_u K_{lqg}) L_g \tilde{u}_p + (I - L_u K_{lqg}) L_h \tilde{e}_p + \tilde{L}_e \tilde{e}_f \tag{5.26}$$

Now that we have derived closed-loop expressions for both u and y, the next step is to calculate their variance expressions which are actually the H_2 norm of the closed-loop expressions weighted by the variance of e. A simple method to derive the variance expression is given below.

Let a disturbance enter the process at time $= t + 1$, i.e.,

$$u_t = u_{t-1} = \dots = u_{t-N+1} = 0$$

$$e_t = e_{t-1} = \dots = e_{t-N+1} = 0$$

$$e_{t+2} = e_{t+3} = \dots = e_{t+N} = 0$$

Then the cumulative effect of the noise e_{t+1} on the process input and output variances can be obtained from equations (5.24) and (5.26), which simplify to

$$u_f = -K_{lqg} l_e e_{t+1} = \begin{bmatrix} \psi_0 \\ \psi_1 \\ \dots \\ \psi_{N-1} \end{bmatrix} e_{t+1} \tag{5.27}$$

$$y_f = (I - L_u K_{lqg}) l_e e_{t+1} = \begin{bmatrix} \gamma_0 \\ \gamma_1 \\ \dots \\ \gamma_{N-1} \end{bmatrix} e_{t+1} \tag{5.28}$$

where $l_e = \begin{bmatrix} L_0 \\ \dots \\ L_{N-1} \end{bmatrix}$, the vector of noise model impulse response coefficients/ Markov parameters. From the above equations, we can calculate the LQG-benchmark variances of the process input and output as

$$Var[u_t] = \sum_{i=0}^{N-1} \psi_i \; Var[e_t] \; \psi_i^T \tag{5.29}$$

$$Var[y_t] = \sum_{i=0}^{N-1} \gamma_i \; Var[e_t] \; \gamma_i^T \tag{5.30}$$

As can be seen from the above equations only the subspace matrices L_u and L_e are required for obtaining the LQG-benchmark variances of the process input and output. The state subspace matrix, L_w, is not required. Therefore the closed loop subspace identification method presented in chapter 3 can be used for obtaining the optimal LQG control variances of process input and output.

For obtaining the LQG-benchmark limit curve, define

$$u_{lqg} = trace\{ \; Var[u_t] \; \} \tag{5.31}$$

$$y_{lqg} = trace\{ \; Var[y_t] \; \} \tag{5.32}$$

For different values of λ, the values for u_{lqg} and y_{lqg} are obtained. A plot of u_{lqg} vs. y_{lqg} represents the optimal LQG performance limit curve that can be used for controller performance assessment.

5.4 Profit analysis of feedforward control

For the case of measured disturbance variables, obtaining optimal benchmark variances helps us in analyzing two things: (i) performance assessment of an existing

feedforward plus feedback controller and (*ii*) profit analysis of implementing a feedforward controller on the process. This analysis in terms of process output variance using MVC-benchmark is provided in [42, 51, 60, 86] (also refer to chapter 3 in this thesis) . In this section we provide the analysis, in terms of both process output variance and process input variance, using the LQG-benchmark.

Consider the case when measurements of some of the disturbance variables, v_t $(h \times 1)$, are available where v_t is assumed to be white noise; if this assumption is not true, then pre-whitening is needed. The process state space representation (2.1)-(2.2) is modified to include measured disturbances as

$$x_{k+1} = Ax_k + \begin{bmatrix} B & B_v \end{bmatrix} \begin{bmatrix} u_k \\ v_k \end{bmatrix} + Ke_k \tag{5.33}$$

$$y_k = Cx_k + \begin{bmatrix} D & D_v \end{bmatrix} \begin{bmatrix} u_k \\ v_k \end{bmatrix} + e_k \tag{5.34}$$

Similarly, the predictor equations for the process output can be expressed as

$$y_f = \Gamma_N^b x_{t+1} + H_N u_f + H_N^v v_f + H_N^s e_f \tag{5.35}$$

$$= L_w^b w_p^b + L_u u_f + L_v v_f + L_e e_f \tag{5.36}$$

where

$$v_p = \begin{bmatrix} v_{t-N+1} \\ \cdots \\ v_t \end{bmatrix}; \quad v_f = \begin{bmatrix} v_{t+1} \\ \cdots \\ v_{t+N} \end{bmatrix}; \quad W_p^b = \begin{bmatrix} Y_p \\ U_p \\ V_p \end{bmatrix} \tag{5.37}$$

The optimal LQG control law, as $N \longrightarrow \infty$, for feedback plus feedforward control, modifies to

$$u_f = -(\lambda I_{Nl} + L_u^T L_u)^{-1} L_u^T L_w^b w_p^b \tag{5.38}$$

$$= -C_{lqg}^b \ \hat{x}_{t+1} \tag{5.39}$$

where

$$C_{lqg}^b = (\lambda I_{Nl} + L_u^T L_u)^{-1} L_u^T \Gamma_N^b \tag{5.40}$$

The relation $L_w^b w_p^b = \Gamma_N^b \hat{x}_{t+1}$ follows from the comparison between subspace equations (5.35) and (5.36). Note that L_v is not required in the design of the

controller and hence not required to be calculated for implementing the controller. However L_v is required for obtaining the LQG benchmark.

Similar to the previous section define

$$L_{g^v} = \begin{bmatrix} G_0^v & G_1^v & ... & G_{N-1}^v & G_N^v \\ G_1^v & G_2^v & ... & G_N^v & 0 \\ ... & ... & ... & ... & ... \\ G_{N-1}^v & 0 & 0 & ... & 0 \end{bmatrix}; \quad \tilde{v}_p = \begin{bmatrix} v_{t+1} \\ v_t \\ ... \\ v_{t-N+1} \end{bmatrix}; \tag{5.41}$$

$$\tilde{L}_v = \begin{bmatrix} 0 & 0 & ... & 0 \\ G_0^v & 0 & ... & 0 \\ ... & ... & ... & ... \\ G_N^v & G_{N-1}^v & ... & 0 \end{bmatrix}; \quad \tilde{v}_f = \begin{bmatrix} v_{t+2} \\ v_{t+3} \\ ... \\ v_{t+N+1} \end{bmatrix} \tag{5.42}$$

where G_i^v is the i-th impulse response coefficient (Markov parameter for multivariate systems) of the disturbance model corresponding to v_t. L_{g^v} can be formed from the subspace matrix L_v. Equations (5.38) and (5.36) can be written as

$$u_f = -K_{lqg}\{L_g u_p + L_{g^v} v_p + L_h e_p\} \tag{5.43}$$

$$y_f = (I - L_u K_{lqg}) L_g \tilde{u}_p + (I - L_u K_{lqg}) L_{g^v} \tilde{v}_p + (I - L_u K_{lqg}) L_h \tilde{e}_p + \tilde{L}_v \tilde{v}_f + \tilde{L}_e \tilde{e}_f \tag{5.44}$$

Consider the measured and unmeasured disturbances enter the process at time = $t+1$, i.e.,

$$\begin{aligned} u_t &= u_{t-1} = ... = u_{t-N+1} = 0 \\ v_t &= v_{t-1} = ... = v_{t-N+1} = 0 \\ e_t &= e_{t-1} = ... = e_{t-N+1} = 0 \\ v_{t+2} &= v_{t+3} = ... = v_{t+N} = 0 \\ e_{t+2} &= e_{t+3} = ... = e_{t+N} = 0 \end{aligned}$$

Therefore

$$u_f = -K_{lqg} l_v v_t - K_{lqg} l_e e_t = \begin{bmatrix} \omega_0 \\ \omega_1 \\ ... \\ \omega_{N-1} \end{bmatrix} v_{t+1} + \begin{bmatrix} \psi_0 \\ \psi_1 \\ ... \\ \psi_{N-1} \end{bmatrix} e_{t+1} \tag{5.45}$$

$$y_f = (I - L_u K_{lqg}) l_v v_t + (I - L_u K_{lqg}) l_e e_t = \begin{bmatrix} \Upsilon_0 \\ \Upsilon_1 \\ \dots \\ \Upsilon_{N-1} \end{bmatrix} v_{t+1} + \begin{bmatrix} \gamma_0 \\ \gamma_1 \\ \dots \\ \gamma_{N-1} \end{bmatrix} e_{t+1} \quad (5.46)$$

where $l_v = \begin{bmatrix} G_0^v \\ \dots \\ G_{N-1}^v \end{bmatrix}$ and $l_e = \begin{bmatrix} L_0 \\ \dots \\ L_{N-1} \end{bmatrix}$, the vectors of noise model impulse response coefficients/ Markov parameters of measured and unmeasured disturbances respectively. From the above equations, we can calculate the LQG-benchmark variances of the process input and output as

$$Var[u_t] = \sum_{i=0}^{N-1} \omega_i \; Var[v_t] \; \omega_i^T + \sum_{i=0}^{N-1} \psi_i \; Var[e_t] \; \psi_i^T \quad (5.47)$$

$$Var[y_t] = \sum_{i=0}^{N-1} \Upsilon_i \; Var[v_t] \; \Upsilon_i^T + \sum_{i=0}^{N-1} \gamma_i \; Var[e_t] \; \gamma_i^T \quad (5.48)$$

Hence only the subspace matrices L_u, L_v and L_e are required for obtaining the LQG-benchmark variances for the process input and output. Now,

$$u_{lqg} = trace\{ \; Var[u_t] \; \} \quad (5.49)$$

$$y_{lqg} = trace\{ \; Var[y_t] \; \} \quad (5.50)$$

By plotting u_{lqg} vs. y_{lqg} for different values of λ, as explained in the previous section, an LQG feedforward plus feedback controller limit curve can be plotted. It can be compared with the feedback-only optimal LQG performance limit curve to analyze the benefits of implementing feedforward control. Further discussion is provided in the next section.

5.5 Controller performance analysis

One of the advantages of LQG-benchmark is that the controller performance can be assessed in terms of both process response (output) variance and the process input variance. The LQG tradeoff curve in figure 5.1 represents the limit of controller

performance, in terms of process input and output variances [8]. That is to say, all linear controllers (from PID, MPC, to any advanced control) can only operate in the region above the curve. Several useful performance indices in the analysis of the controller performance can be obtained from the LQG-benchmark curve.

5.5.1 Case 1: Feedback controller acting on the process with no measured disturbances

Consider the case when a feedback-only controller is acting on the process and the actual input and output variances are represented as $(V_u)^{fb}$ and $(V_y)^{fb}$ respectively. The closer $(V_u)^{fb}$ and $(V_y)^{fb}$ are to the limit curve, the closer is the controller performance to the optimal LQG controller. If the optimal output variance corresponding to $(V_u)^{fb}$ is $(V_y^o)^{fb}$ and the optimal input variance corresponding to $(V_y)^{fb}$ is $(V_u^o)^{fb}$, then the LQG performance indices can be defined in terms of process response variance, $(\eta)^{fb}$, and process input variance, $(E)^{fb}$, as

$$(\eta)^{fb} = \frac{(V_y^o)^{fb}}{(V_y)^{fb}}; \qquad (E)^{fb} \triangleq \frac{(V_u^o)^{fb}}{(V_u)^{fb}} \tag{5.51}$$

$(\eta)^{fb}$ and $(E)^{fb}$ vary between 0 and 1. If $(\eta)^{fb}$ is equal to 1, for the given input variance, then the controller is giving optimal performance with respect to the process variance. If not, then the controller is non-optimal and there is scope for improvement in terms of process response. Similarly if $(E)^{fb}$ is equal to 1, for the given output variance, then the controller is giving optimal performance with respect to the input variance. If not, then the controller is non-optimal and there is scope to reduce input variance.

The maximum possible % improvement in controller performance with respect to process response variance without increasing the input variance, by retuning the controller, can be calculated as

$$(I_\eta)^{fb} = \frac{(V_y)^{fb} - (V_y^o)^{fb}}{(V_y)^{fb}} \ 100\% \tag{5.52}$$

Similarly the maximum possible % improvement in controller performance with respect to input variance without increasing the output variance, by retuning the

controller, can be calculated as

$$(I_E)^{fb} = \frac{(V_u)^{fb} - (V_u^o)^{fb}}{(V_u)^{fb}} \ 100\% \tag{5.53}$$

5.5.2 Case 2: Feedforward plus feedback controller acting on the process

For the case of measured disturbances, and a feedforward plus feedback controller acting on the process, let the actual input and output variances be denoted by $(V_u)^{ff\&fb}$ and $(V_y)^{ff\&fb}$ respectively. Then the LQG curve is plotted using $\{L_u, L_e$, and $L_v\}$[2] and represents the limit of performance in terms of process input and output variances for a feedforward plus feedback controller. Let the optimal output variance corresponding to $(V_u)^{ff\&fb}$ be $(V_y^o)^{ff\&fb}$ and the optimal input variance corresponding to $(V_y)^{ff\&fb}$ be $(V_u^o)^{ff\&fb}$, then the optimal FF-FB LQG performance indices can be defined in terms of process response variance, $(\eta)^{ff\&fb}$, and process input variance, $(E)^{ff\&fb}$, as

$$(\eta)^{ff\&fb} = \frac{(V_y^o)^{ff\&fb}}{(V_y)^{ff\&fb}}; \qquad (E)^{ff\&fb} = \frac{(V_u^o)^{ff\&fb}}{(V_u)^{ff\&fb}} \tag{5.54}$$

$(\eta)^{ff\&fb}$ and $(E)^{ff\&fb}$ vary between 0 and 1. If $(\eta)^{ff\&fb}$ is equal to 1, then the controller is giving optimal feedforward plus feedback controller performance, for the given input variance. If not, then the controller is non-optimal and has potential for improvement by retuning. Similarly if $(E)^{ff\&fb}$ is equal to 1, then the controller is giving optimal feedforward plus feedback controller performance, for the given output variance.

The maximum possible % improvement in the controller performance, with respect to process response variance without increasing the input variance, by retuning the controller, is calculated for the feedforward plus feedback control case as

$$(I_\eta)^{ff\&fb} = \frac{(V_y)^{ff\&fb} - (V_y^o)^{ff\&fb}}{(V_y)^{ff\&fb}} \ 100\% \tag{5.55}$$

[2]It should be noted that the subspace matrix corresponding to the measured disturbances, L_v, cannot be identified when a feedforward plus feedback controller is acting on the process [58]. A feedback-only controller should be acting on the process for identifying L_v.

Similarly we can define in terms of the input variance

$$(I_E)^{ff\&fb} = \frac{(V_u)^{ff\&fb} - (V_u^o)^{ff\&fb}}{(V_u)^{ff\&fb}} \ 100\% \tag{5.56}$$

5.5.3 Case 3: Feedback controller acting on the process with measured disturbances

Consider the case where a feedback-only controller is acting on the process and measured disturbance variables are available. We want to know how much improvement in the controller performance is possible by implementing a feedforward control in addition to the existing feedback-only controller. By implementing optimal feedforward control on the system the process response variance will decrease. The same may not hold for the input variance. The process input variance may increase or decrease by the implementation of feedforward control with the measured disturbances. The following analysis helps in determining the incentive for the implementation of a feedforward controller on the process.

We can obtain $(V_u)^{fb}$ and $(V_y)^{fb}$ from process data. We construct two LQG limit curves. (i) Identify $\{L_u$, L_v, and $L_e\}$ and construct the FF & FB LQG controller limit curve to obtain $(V_y^o)^{ff\&fb}$ and $(V_u^o)^{ff\&fb}$. (ii) Treat measured and unmeasured disturbances as a lumped set of $(m \times 1)$ unmeasured disturbances and identify $\{L_u$ and $L_e\}$ and construct the FB-only LQG controller limit curve to obtain $(V_y^o)^{fb}$ and $(V_u^o)^{fb}$.

The maximum possible improvement in the optimal controller performance with the implementation of an optimal feedforward controller is obtained in terms of the process response variance and process input variance as

$$\frac{(V_y^o)^{fb} - (V_y^o)^{ff\&fb}}{(V_y)^{fb}} \ 100\% \tag{5.57}$$

and

$$\frac{(V_u^o)^{fb} - (V_u^o)^{ff\&fb}}{(V_u)^{fb}} \ 100\% \tag{5.58}$$

respectively.

The performance analysis indices presented for three different cases above can be used in analyzing the incentives, in terms of decreasing both process response variance and process input variance, for retuning the controller.

5.6 Summary of the subspace matrices approach to the calculation of LQG-benchmark

Controller performance analysis using LQG-benchmark involves comparing the current process input and output variances with the variances if an LQG controller were implemented on the process. The method proposed in this chapter allows the calculation of the LQG-benchmark variances directly from the deterministic and stochastic process subspace matrices, thus not requiring a parametric model, and principally consists of the following steps:

1. Estimation for the deterministic and stochastic process subspace matrices from the process data. The subspace matrices can be identified by either

(a) Using the process open loop data [91, 93] as shown in chapter 2.

(b) Using the process closed loop data with setpoint excitation [58] as shown in chapter 3.

2. Estimation of the process stochastic noise and obtaining the variance, $Var[e_t]$. Also estimate $Var[v_t]$ if any measured disturbances are available; if the measured disturbance is not white noise, then pre-whitening is necessary. Routine operating data can be used for this purposes.

3. For different values of λ calculate the LQG-benchmark variances, u_{lqg} and y_{lqg}. Plot u_{lqg} vs. y_{lqg} to obtain the optimal LQG performance limit curve.

4. For the current process input and output variances, V_u and V_y respectively, obtain the optimal variance values V_u^o and V_y^o, for both feedback-only and feedforward plus

feedback control cases. Calculate the controller performance analysis indices.

$$\begin{array}{ll} (\eta)^{fb} & (E)^{fb} \\ (I_\eta)^{fb} & (I_E)^{fb} \\ (\eta)^{ff\&fb} & (E)^{ff\&fb} \\ (I_\eta)^{ff\&fb} & (I_E)^{ff\&fb} \end{array}$$

5.7 Simulations

Consider the following state space system (modified from the example in [93])

$$\begin{aligned} x_{k+1} &= \begin{bmatrix} 0.6 & 0.6 & 0 \\ -0.6 & 0.6 & 0 \\ 0 & 0 & 0.7 \end{bmatrix} x_k + \begin{bmatrix} 1.6161 \\ -0.3481 \\ 2.6319 \end{bmatrix} u_k + \begin{bmatrix} 0.5 \\ -0.5 \\ 0.4 \end{bmatrix} v_k + \begin{bmatrix} -1.1472 \\ -1.5204 \\ -3.1993 \end{bmatrix} e_k \\ y_k &= \begin{bmatrix} -0.4373 & -0.5046 & 0.0936 \end{bmatrix} x_k + [-0.7759] u_k + [-0.5] v_k + e_k \end{aligned}$$

A process time delay of 3-samples is introduced for the above system in MATLAB-*Simulink*. A PID controller, $[0.1 + \frac{0.05}{s} + 0.05s]$, is tuned for the system. We assume the controller knowledge is not known in the following analysis.

Closed loop input/output data is obtained by exciting the system using a designed 'RBS' signal (with *idinput* function in MATLAB), with bandpass limits $\begin{bmatrix} 0 & 0.06 \end{bmatrix}$ and magnitude 1, for the setpoint. The measured disturbance and the unmeasured disturbance are random white noise with standard deviations 0.2 and 0.1 respectively. Note that although the measured and unmeasured disturbances can have same standard deviation, different *seeds* have to be used for generating them in MATLAB. Using closed loop subspace identification [58], with $N = 30$ (rows) and $j = 3000$ (columns) in the data Hankel matrices, the subspace matrices L_u (30×30) and L_e (30×30) are identified. Due to the presence of noise, the upper non-diagonal elements in L_u and L_e will not be exactly zero but negligibly small numbers (they approach to zero as $N \longrightarrow \infty$). An optimal feedback-only LQG controller is considered as a benchmark for controller performance assessment. For a range of values of λ $(1 - 30)$, the LQG-benchmark variances of the process input

and output are obtained for both feedback-only control case and feedforward plus feedback control case and plotted in figure 5.2. The controller performance analysis parameters are obtained from figure 5.2 as:

parameter		*value*	*parameter*		*value*
$(V_y)^{fb}$	.....	0.0549			
$(V_u)^{fb}$	.....	13×10^{-4}			
$(V_y^o)^{fb}$	.....	0.0506	$(V_y^o)^{ff\&fb}$	.....	0.0474
$(V_u^o)^{fb}$	.....	1.8×10^{-4}	$(V_u^o)^{ff\&fb}$	.....	8.2×10^{-4}
$(\eta)^{fb}$	.....	0.9217	$(\eta)^{ff\&fb}$	.....	0.8634
$(I_\eta)^{fb}$	.....	07.83%	$(I_\eta)^{ff\&fb}$	.....	13.66%
$(E)^{fb}$	.....	0.1385	$(E)^{ff\&fb}$	.....	0.6308
$(I_E)^{fb}$	.....	86.15%	$(I_\eta)^{ff\&fb}$	.....	36.92%
			$\frac{(V_y^o)^{fb}-(V_y^o)^{ff\&fb}}{(V_y)^{fb}}100\%$	.....	05.83%
			$\frac{(V_u^o)^{fb}-(V_u^o)^{ff\&fb}}{(V_u)^{fb}}100\%$	.....	-49.23%

Table 5.1: LQG-benchmark performance assessment parameters for the simulations.

We make the following observations from table (5.1):

From feedback-only LQG limit curve

From the FB-only optimal LQG-benchmark variances we see that although the controller performance is close to optimal with respect to the process output variance (92.17 % of the optimal), the performance index with respect to the input variability is only 13.85%. Hence there is a maximum possible scope of 86.15% to reduce the input variance without increasing the output variance.

From feedforward & feedback LQG limit curve

From the FF&FB optimal LQG-benchmark variances we see that the controller performance is still close to optimal with respect to the process response variance (86.34 % of the optimal), whereas the performance index with respect to the input variability is better than with a feedback-only controller 63.08%. Hence there is a maximum possible scope of 36.92% to reduce the input variance without increasing the output variance.

For the profit analysis of implementation of feedforward controller on the process, we see that there is an incentive of only 5.83% reduction in the process response variance possible by the implementation of an optimal feedforward controller on the process. However, there is 49.23% maximum possible scope for decrease in the process input variance. Hence it can be concluded that there is not much incentive for the implementation of a feedforward controller in this case.

5.8 Application on a pilot scale process

The proposed method of controller performance analysis using optimal LQG-benchmark is tested on a pilot scale system. The system considered is shown in figure (5.5). The input (u) is the valve position of the input water flow valve and the process variable to be controlled (y) is the level of water in the tank. The tank outlet flow valve is kept at a constant position. The head of the water in the inlet pipe can be considered as (an unmeasured) disturbance. The tank level is controlled by a PID controller, $5 + \frac{0.05}{s} + 0.05s$. The controller sampling interval is 5 seconds. An '*RBS*' signal of series of setpoint changes to the level is designed in *MATLAB*. Closed loop data of the process input and output is collected at a sampling rate of 5 seconds. The subspace matrices L_u and L_e of dimension 200 are identified using the closed loop subspace identification method from chapter 3. The impulse response coefficients models for the process and noise are plotted in figure 5.6. The optimal LQG-benchmark curve is plotted for a range of values of λ, as shown in

figure 5.7. The actual process input and output variances are compared with the optimal variances for the controller performance analysis.

parameter		*value*
$(V_y)^{fb}$	.....	1.885×10^{-4}
$(V_u)^{fb}$	.....	1.52×10^{-3}
$(V_y^o)^{fb}$	.....	0.71×10^{-4}
$(V_u^o)^{fb}$	.....	1.66×10^{-4}
$(\eta)^{fb}$	.....	0.378
$(I_\eta)^{fb}$	.....	62.30%
$(E)^{fb}$	.....	0.11
$(I_\eta)^{fb}$	.....	89.00%

Table 5.2: LQG-benchmark performance assessment parameters for the pilot scale process.

From table (5.2) we can see that the controller performance is non-optimal with respect to both input and output variances. There is a maximum possible scope of 62.30% to reduce the process output variance without increasing the input variance and 89.00% to reduce the process input variance without increasing the output variance.

5.9 Conclusions

A subspace identification based approach is proposed in this chapter for obtaining the optimal LQG-benchmark from closed loop data, for controller performance assessment. It has been shown that instead of using explicit process models, only the subspace matrices corresponding to the deterministic and stochastic inputs, L_u and L_e, are required to obtain the LQG-benchmark. The closed loop subspace identification method is used to obtain L_u and L_e which are subsequently used to obtain the LQG-benchmark. The optimal LQG-benchmark method is extended to the case of feedforward plus feedback control. Profit analysis for the implementation of feedforward control under optimal LQG control performance framework is also

derived and explained in this chapter. The results of the chapter are illustrated through a simulation example and a pilot-scale experiment.

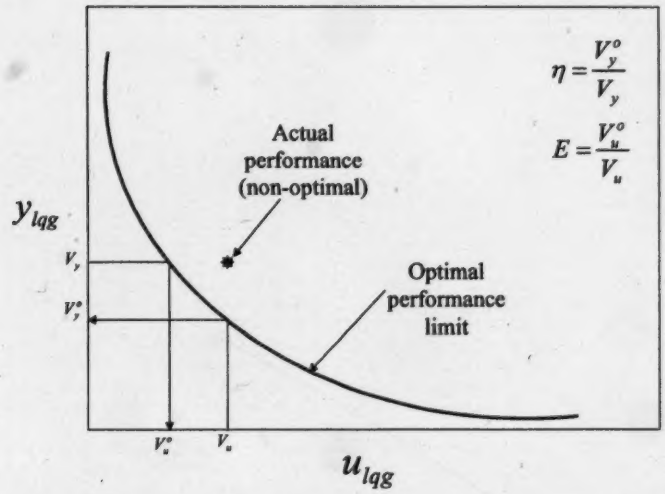


Figure 5.1: Optimal LQG control performance limit curve. V_u and V_y represent the variances obtained from process data while V_u^o and V_y^o represent the optimal LQG-benchmark variances.

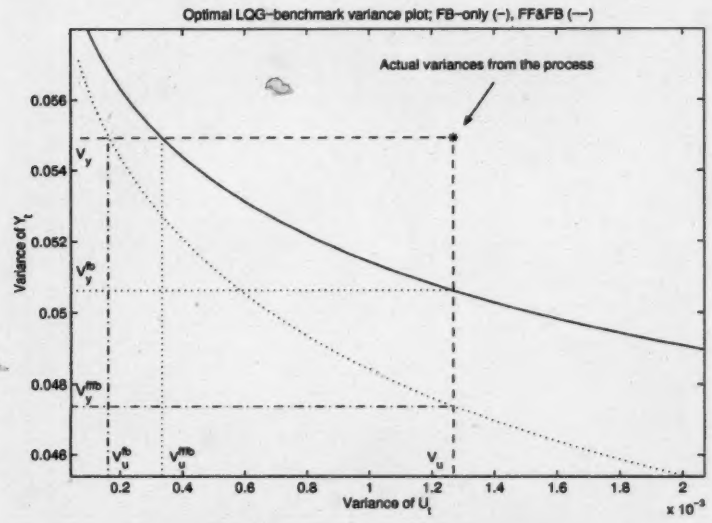


Figure 5.2: Optimal LQG control performance limit curve.

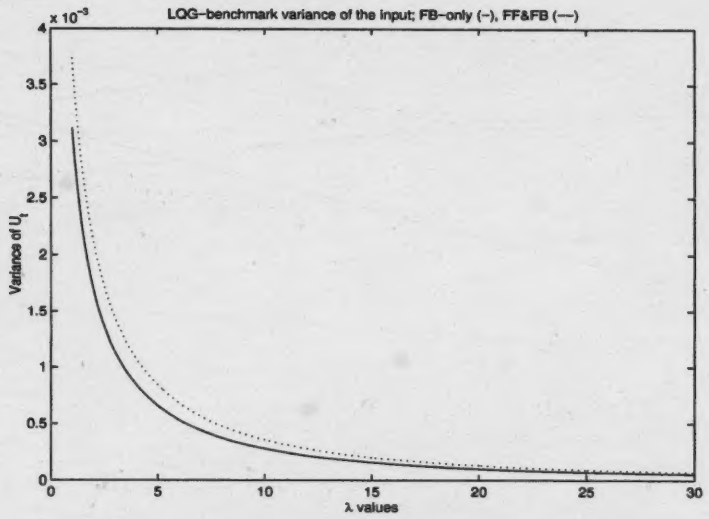


Figure 5.3: LQG-benchmark variances of the input.

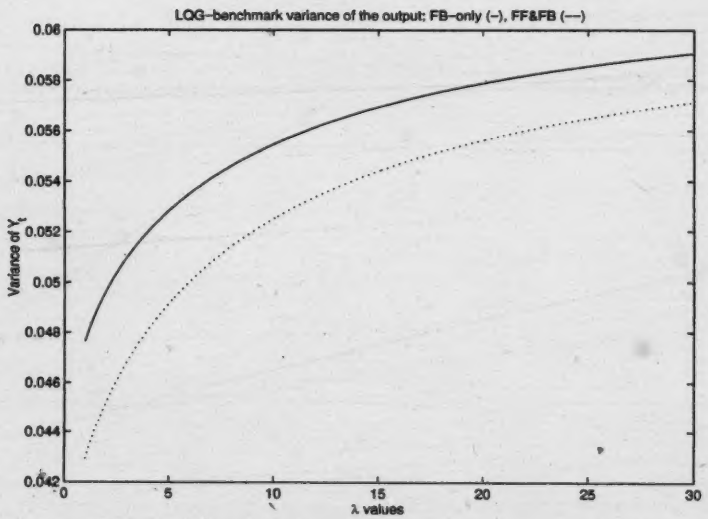


Figure 5.4: LQG-benchmark variances of the output.

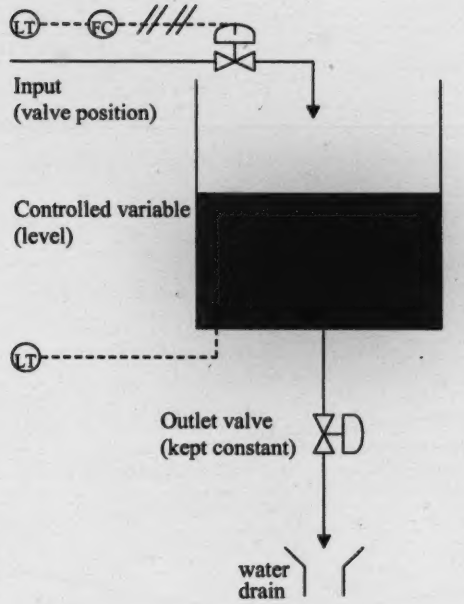


Figure 5.5: Experimental setup

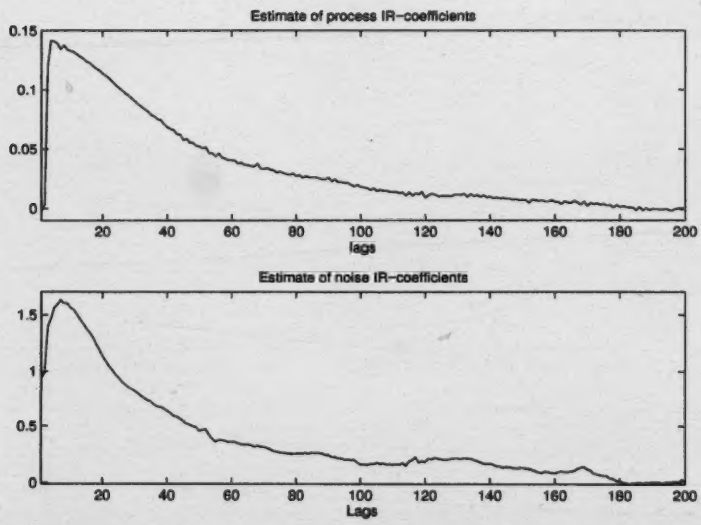


Figure 5.6: Impulse response models for the process and noise identified using closed loop data

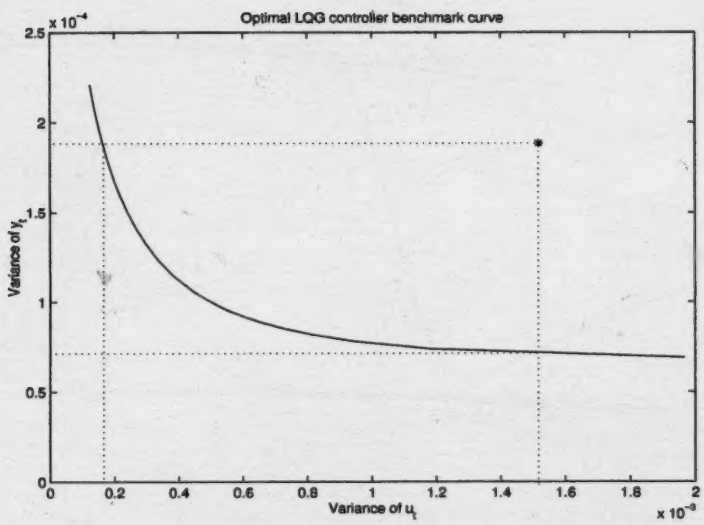


Figure 5.7: Optimal LQG-benchmark curve for the CSTH

Chapter 6

Estimation of the statistical confidence intervals for LQG-benchmark obtained using the subspace matrices

6.1 Introduction

In chapter 5, the expressions for obtaining the LQG-benchmark variances for the process inputs and outputs are derived for controller performance assessment. Industrial data inherently has noise and this noise transfers to the subspace matrices identified for the LQG-benchmark variances calculation. In this chapter we derive the expressions for calculating the confidence intervals for the LQG-benchmark variances calculated using the subspace matrices based method. We use extensively the Kronecker algebra and matrix calculus to derive these expressions.

6.2 Kronecker algebra and matrix calculus

Differentiation of matrices functions involves Kronecker algebra. Consider a vector $p = [p_i]_{q\times 1}$ and matrices $A = [a_{ij}]_{\alpha\times\beta}$, $B = [b_{ij}]_{\sigma\times\rho}$. Assume $(\alpha > \beta)$. The following rules of operations and differentiation are useful for the statistical analysis provided in this chapter. Most of the formulae presented in this section are directly taken from [9, 31, 123, 124, 126], without presenting the detailed derivations involved.

6.2.1 Kronecker algebra

$$A \otimes B = matrix[a_{ij}B] = \begin{bmatrix} a_{11}B & a_{12}B & \dots & a_{1\beta}B \\ a_{21}B & a_{22}B & \dots & a_{2\beta}B \\ \dots & \dots & \dots & \dots \\ a_{\alpha 1}B & a_{\alpha 2}B & \dots & a_{\alpha\beta}B \end{bmatrix}_{\alpha\sigma\times\beta\rho}$$

$$\begin{aligned} A\otimes(B+C) &= A\otimes B + A\otimes C \\ (A\otimes B)(C\otimes D) &= (AC)\otimes(BD) \\ (A\otimes B)^T &= A^T\otimes B^T \\ (A\otimes B)^{-1} &= A^{-1}\otimes B^{-1} \\ (A\otimes B)^{\dagger} &= A^{\dagger}\otimes B^{\dagger} \qquad (\dagger\ represents\ pseudo-inverse) \\ (I_s\otimes B)^{\sigma} &= I_s\otimes B^{\sigma} \\ A\otimes B &= U_1(B\otimes A)U_2 \qquad (U_1\ \&\ U_2\ are\ permutation\ matrices) \end{aligned}$$

where $U_{\alpha\beta}^{\alpha\beta\times\alpha\beta}$ is a permutations matrix defined as

$$\begin{aligned} U_{\alpha\beta}^{\alpha\beta\times\alpha\beta} &= \sum_{i=1}^{\alpha}\sum_{j=1}^{\beta} E_{ij}^{(\alpha\times\beta)}\otimes(E_{ij}^{(\alpha\times\beta)})^T \\ \bar{U}_{\alpha\beta}^{\alpha^2\times\beta^2} &= \sum_{i=1}^{\alpha}\sum_{j=1}^{\beta} E_{ij}^{(\alpha\times\beta)}\otimes E_{ij}^{(\alpha\times\beta)} \\ U_{\alpha\beta}^{-1} &= U_{\alpha\beta}^{T} = U_{\alpha\beta} \qquad (symmetric\ and\ orthogonal) \end{aligned}$$

with $E_{ij}^{\alpha\times\beta}$ being the $(\alpha\times\beta)$ Kronecker matrix which has '1' in the i,j-element and '0' elsewhere.

6.2.2 Differentiating a matrix or a matrix function w.r.t. a matrix

$$\begin{aligned}
\frac{\partial A}{\partial B} &= \begin{bmatrix} \frac{\partial A}{\partial b_{11}} & \frac{\partial A}{\partial b_{12}} & \cdots & \frac{\partial A}{\partial b_{1l}} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial A}{\partial b_{\sigma 1}} & \frac{\partial A}{\partial b_{\sigma 2}} & \cdots & \frac{\partial A}{\partial b_{\sigma l}} \end{bmatrix} = \sum_{i=1}^{\sigma}\sum_{j=1}^{\rho} E_{ij} \otimes \frac{\partial A}{\partial b_{ij}} \\
\left[\frac{\partial A}{\partial B}\right]^T &= \frac{\partial A^T}{\partial B^T} \\
\frac{\partial A}{\partial A} &= \bar{U}_{\alpha\beta}^{\alpha^2 \times \beta^2} \\
\frac{\partial A^T}{\partial A} &= U_{\alpha\beta}^{\alpha\beta \times \alpha\beta} \\
\frac{\partial A^{-1}}{\partial A} &= -(I_\alpha \otimes A^{-1})\frac{\partial A}{\partial A}(I_\beta \otimes A^{-1}) \\
\frac{\partial AB}{\partial M} &= \frac{\partial A}{\partial M}(I_s \otimes B) + (I_r \otimes A)\frac{\partial B}{\partial M} \\
\frac{\partial [A \otimes B]}{\partial M} &= \frac{\partial A}{\partial M} \otimes B + (I_r \otimes U_{\alpha\sigma}^{\alpha\sigma \times \alpha\sigma})(\frac{\partial B}{\partial M} \otimes A)(I_s \otimes U_{\beta l}^{\beta l \times \beta l})
\end{aligned}$$

where $M = [M_{ij}]_{r \times s}$ and, A and B have compatible dimensions.

6.3 Statistical analysis for matrices

When dealing with matrices instead of vectors or scalars, the extra dimension needs to be carefully incorporated in the derivations. Consider a matrix

$$A = \begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1\beta} \\ a_{21} & a_{22} & \ldots & a_{2\beta} \\ \ldots & \ldots & \ldots & \ldots \\ a_{\alpha 1} & a_{\alpha 2} & \ldots & a_{\alpha\beta} \end{bmatrix} = \begin{bmatrix} \mathbf{a_1} & | & \mathbf{a_2} & | & \ldots & | & \mathbf{a_\beta} \end{bmatrix}_{\alpha \times \beta}$$

The covariance matrix for the matrix A can be defined as

$$\begin{aligned}
\Sigma_A &= E\left[(A - \hat{A}) \otimes (A - \hat{A})^T\right] \\
&= \begin{bmatrix} cov(\mathbf{a_1}, \mathbf{a_1}) & cov(\mathbf{a_2}, \mathbf{a_1}) & \ldots & cov(\mathbf{a_\beta}, \mathbf{a_1}) \\ cov(\mathbf{a_1}, \mathbf{a_2}) & cov(\mathbf{a_2}, \mathbf{a_2}) & \ldots & cov(\mathbf{a_\beta}, \mathbf{a_1}) \\ \ldots & \ldots & \ldots & \ldots \\ cov(\mathbf{a_1}, \mathbf{a_\beta}) & cov(\mathbf{a_2}, \mathbf{a_\beta}) & \ldots & cov(\mathbf{a_\beta}, \mathbf{a_\beta}) \end{bmatrix}_{\alpha\beta \times \alpha\beta}
\end{aligned}$$

where

each of the block $cov(\mathbf{a_i}, \mathbf{a_j}) = \begin{bmatrix} cov(a_{1i}, a_{1j}) & cov(a_{2i}, a_{1j}) & ... & cov(a_{\alpha i}, a_{1j}) \\ cov(a_{1i}, a_{2j}) & cov(a_{2i}, a_{2j}) & ... & cov(a_{\alpha i}, a_{1j}) \\ ... & ... & ... & ... \\ cov(a_{1i}, a_{\alpha j}) & cov(a_{2i}, a_{\alpha j}) & ... & cov(a_{\alpha i}, a_{\alpha j}) \end{bmatrix}$ is an $(\alpha \times \alpha)$ covariance matrix. To avoid complex mathematical formulation, the functions of matrices can be equivalently expressed as functions of vectors formed by the columns/rows of the matrices. This will allow us to use the vector functions' Taylor series expansion equations for the functions of matrices. We can express the matrix A as a column vector, $\bar{a}$, defined as $\bar{a} = \begin{bmatrix} \mathbf{a_1} \\ \mathbf{a_2} \\ ... \\ \mathbf{a_\beta} \end{bmatrix}_{\alpha\beta \times 1}$. In MATLAB, the function is $\bar{a} = A(:);$.

6.3.1 Taylor series expansion

If a matrix $F(\alpha \times \beta)$ is a function of a vector, $p(q \times 1)$, then

$$F(p) = F(p_0) + \sum_{i=1}^{\alpha} \frac{1}{i!} [(p - p_0)^{T[i]} \otimes I_\alpha] \left(\frac{\partial^i}{\partial p^i} F(p) \mid_{p=p_0} \right) \tag{6.1}$$

$$= F(p_0) + \sum_{i=1}^{\alpha} \frac{1}{i!} \left(\frac{\partial^i}{\partial p^{Ti}} F(p) \mid_{p=p_0} \right) [(p - p_0)^{[i]} \otimes I_\beta] \tag{6.2}$$

with

$$(p - p_0)^{T[i]} = (p - p_0)^T \otimes (p - p_0)^T \otimes ... \otimes (p - p_0)^T \tag{6.3}$$

6.3.2 Basic derivations

We can derive

$$\frac{\partial A}{\partial \bar{a}} = \bar{V}_{\alpha\beta}^{\alpha^2\beta \times \beta} \tag{6.4}$$

$$\frac{\partial A^T}{\partial \bar{a}} = V_{\alpha\beta}^{\alpha\beta^2 \times \alpha} \tag{6.5}$$

where

$$\bar{V}_{\alpha\beta}^{\alpha^2\beta\times\beta} = \sum_{i=1}^{\alpha}\sum_{j=1}^{\beta} O_{ij}^{(\alpha\beta\times 1)} \otimes E_{ij}^{(\alpha\times\beta)} \tag{6.6}$$

$$V_{\alpha\beta}^{\alpha\beta^2\times\alpha} = \sum_{i=1}^{\alpha}\sum_{j=1}^{\beta} O_{ij}^{(\alpha\beta\times 1)} \otimes (E_{ij}^{(\alpha\times\beta)})^T \tag{6.7}$$

with $O_{ij}^{\alpha\beta\times 1}$ being the $(\alpha\beta \times 1)$ column matrix which has '1' in the $[\alpha(j-1)+i]^{th}$-element and '0' elsewhere.

We can derive the expression for $\frac{\partial A^{-1}}{\partial \bar{a}}$, $\frac{\partial (A^T)^{-1}}{\partial \bar{a}}$, $\frac{\partial A^\dagger}{\partial \bar{a}}$ and $\frac{\partial (A^T)^\dagger}{\partial \bar{a}}$ as follows:

(*i*) $\quad \frac{\partial A^{-1}}{\partial \bar{a}} = -\left[I_{\alpha^2} \otimes \overline{A^{-1}}\right] \bar{V}_{\alpha^2}^{\alpha^3\times\alpha} A^{-1}$

Proof: $\alpha = \beta$ is assumed for A^{-1} to exist.

$$\begin{aligned} AA^{-1} &= I \\ \frac{\partial (AA^{-1})}{\partial \bar{a}} &= \frac{\partial A}{\partial \bar{a}} A^{-1} + [I_{\alpha^2} \otimes A] \frac{\partial A^{-1}}{\partial \bar{a}} = 0 \\ \frac{\partial A^{-1}}{\partial \bar{a}} &= -[I_{\alpha^2} \otimes A]^{-1} \frac{\partial A}{\partial \bar{a}} A^{-1} \\ &= -\left[I_{\alpha^2} \otimes A^{-1}\right] \bar{V}_{\alpha^2}^{\alpha^3\times\alpha} A^{-1} \end{aligned} \tag{6.8}$$

(*ii*) $\quad \frac{\partial (A^T)^{-1}}{\partial \bar{a}} = -\left[I_{\alpha^2} \otimes A^T\right]^{-1} \frac{\partial A^T}{\partial \bar{a}} (A^T)^{-1} = -\left[I_{\alpha^2} \otimes (A^T)^{-1}\right] V_{\alpha^2}^{\alpha^3\times\alpha} (A^T)^{-1}$

Proof: Similar to above.

(*iii*) $\quad \frac{\partial A^\dagger}{\partial \bar{a}} = -\left[I_{\alpha\beta} \otimes A^\dagger\right] \bar{V}_{\alpha\beta}^{\alpha^2\beta\times\beta} A^\dagger$

Proof:

$$\begin{aligned} A^\dagger A A^\dagger &= A^\dagger \\ \frac{\partial (A^\dagger A A^\dagger)}{\partial \bar{a}} &= \frac{\partial A^\dagger}{\partial \bar{a}} (AA^\dagger) + \left[I_{\alpha\beta} \otimes A^\dagger\right] \frac{\partial (AA^\dagger)}{\partial \bar{a}} \\ &= \frac{\partial A^\dagger}{\partial \bar{a}} + \left[I_{\alpha\beta} \otimes A^\dagger\right] \left\{ \frac{\partial A}{\partial \bar{a}} A^\dagger + [I_{\alpha\beta} \otimes A] \frac{\partial A^\dagger}{\partial \bar{a}} \right\} \\ &= \frac{\partial A^\dagger}{\partial \bar{a}} \end{aligned}$$

Therefore,

$$\frac{\partial A^\dagger}{\partial \bar{a}} = -[I_{\alpha\beta} \otimes A]^{-1} \frac{\partial A}{\partial \bar{a}} A^\dagger = -\left[I_{\alpha\beta} \otimes A^\dagger\right] \bar{V}_{\alpha\beta}^{\alpha^2\beta\times\beta} A^\dagger \tag{6.9}$$

$$(iv) \qquad \frac{\partial (A^T)^\dagger}{\partial \bar{a}} = -\left[I_{\alpha\beta} \otimes (A^T)^\dagger\right] \frac{\partial A^T}{\partial \bar{a}} (A^T)^\dagger = -\left[I_{\alpha\beta} \otimes (A^T)^\dagger\right] V_{\alpha\beta}^{\alpha\beta^2 \times \alpha} (A^T)^\dagger$$

Proof: Similar to above.

Note the following properties for the vector $\bar{a}$:

$$E[\bar{a} - \hat{\bar{a}}] = 0 \tag{6.10}$$

$$E[(\bar{a} - \hat{\bar{a}}) \otimes (\bar{a} - \hat{\bar{a}})] = S_{\bar{a}} \tag{6.11}$$

where $S_{\bar{a}}$ is a column vector obtained from the covariance matrix, Σ_A (in MATLAB notation, $S_{\bar{a}} = \Sigma_A(:)$). Similarly, $E[(\bar{a} - \hat{\bar{a}})^T \otimes (\bar{a} - \hat{\bar{a}})^T] = S_{\bar{a}}^T$, a row vector. Suppose there is another vector $\left[\bar{b}\right]_{\sigma\rho \times 1}$ formed from a matrix $B = [B_{ij}]_{\sigma \times \rho}$. If $\bar{a}$ is uncorrelated with $\bar{b}$, then

$$E[(\bar{a} - \hat{\bar{a}})^T \otimes (\bar{b} - \hat{\bar{b}})^T] = 0 \tag{6.12}$$

6.4 Statistical properties of matrices as random variables

For the matrix $A = [A_{ij}]_{\alpha \times \beta} = \left[\mathbf{a_1} \mid \mathbf{a_2} \mid \ldots \mid \mathbf{a_\beta} \right]_{\alpha \times \beta}$ where $\mathbf{a_i}$ is a $(\alpha \times 1)$ column vector. To avoid complex mathematical formulation, the functions of matrices can be equivalently expressed as functions of vectors formed by the columns/rows of the matrices. This will allow us to use the vector functions' Taylor series expansion equations for the functions of matrices. For example $f(A)$ can be equivalently expressed as $f(\bar{a})$, where $\bar{a} = \begin{bmatrix} \mathbf{a_1} \\ \mathbf{a_2} \\ \ldots \\ \mathbf{a_\beta} \end{bmatrix}_{\alpha\beta \times 1}$. Note the following properties for the vector $\bar{a}$:

$$E[\bar{a} - \hat{\bar{a}}] = 0 \tag{6.13}$$

$$E[(\bar{a} - \hat{\bar{a}}) \otimes (\bar{a} - \hat{\bar{a}})] = S_{\bar{a}} \tag{6.14}$$

where $S_{\bar{a}}$ is the covariance matrix, Σ_A, arranged as a column vector. Similarly, $E[(\bar{a} - \hat{\bar{a}})^T \otimes (\bar{a} - \hat{\bar{a}})^T] = S_{\bar{a}}^T$, a row vector. Suppose there is another vector $\bar{b}$ formed

from a matrix $B = [B_{ij}]_{\sigma\times\rho}$. If $\bar{a}$ is uncorrelated with $\bar{b}$, then

$$E[(\bar{a} - \hat{\bar{a}}) \otimes (\bar{b} - \hat{\bar{b}})] \;=\; 0 \tag{6.15}$$

Consider the following derivations:

1. $[F_1(A)]_{\alpha\times\beta} = A^T$

$$\frac{\partial F_1}{\partial \bar{a}} \;=\; \frac{\partial A^T}{\partial \bar{a}} = V_{\alpha\beta}^{\alpha\beta\times\alpha\beta} \tag{6.16}$$

The Taylor series expansion for $F_1(\bar{a})$ about the point $\hat{\bar{a}}$ can be written as

$$F_1 \;=\; \hat{A}^T + \left[(\bar{a} - \hat{\bar{a}})^T \otimes I_\alpha\right] \frac{\partial A^T}{\partial \bar{a}} + \ldots \tag{6.17}$$

$$E[A^T] \;=\; \hat{A}^T \tag{6.18}$$

2. $[F_2(A)]_{\alpha\beta\times\alpha\beta} = (A^T - \hat{A}^T) \otimes (A^T - \hat{A}^T)^T$

$$\begin{aligned} F_2(A) \;&=\; (A^T - \hat{A}^T) \otimes (A^T - \hat{A}^T)^T \\ &=\; \left\{(A - \hat{A}) \otimes (A - \hat{A})^T\right\}^T \end{aligned} \tag{6.19}$$

Therefore

$$\Sigma_{A^T} \;=\; E[F_2] = \left\{E[(A - \hat{A}) \otimes (A - \hat{A})^T]\right\}^T \tag{6.20}$$

$$=\; \Sigma_A^T \tag{6.21}$$

Therefore we can write the distribution for A^T as

$$A^T \approx N\left(\hat{A}^T, \Sigma_A\right) \tag{6.22}$$

3. $[F_3(A)]_{\alpha\times\alpha} = A^{-1}$

$\alpha = \beta$ is assumed in derivations 3-4.

$$\frac{\partial F_3}{\partial \bar{a}} \;=\; \frac{\partial A^{-1}}{\partial \bar{a}} = -\left[I_{\alpha^2} \otimes A^{-1}\right] \bar{V}_{\alpha^2}^{\alpha^3\times\alpha} A^{-1} \tag{6.23}$$

The Taylor series expansion for $F_3(A)$ about the point $\hat{\bar{a}}$ can be written as

$$F_3 = \hat{A}^{-1} - \left[(\bar{a}-\hat{\bar{a}})^T \otimes I_\alpha\right]\left[I_{\alpha^2}\otimes A^{-1}\right]\bar{V}_{\alpha^2}^{\alpha^3\times\alpha}A^{-1} + \ldots \quad (6.24)$$

$$E[A^{-1}] = \hat{A}^{-1} \quad (6.25)$$

4. $[F_4(A)]_{\alpha^2\times\alpha^2} = (A^{-1}-\hat{A}^{-1})\otimes(A^{-1}-\hat{A}^{-1})^T$

$$\frac{\partial F_4}{\partial \bar{a}} = \frac{\partial A^{-1}}{\partial \bar{a}}\otimes(A^{-1}-\hat{A}^{-1})^T + \left[I_{\alpha^2}\otimes U_{\alpha^2}^{\alpha^2\times\alpha^2}\right]\left[\frac{\partial (A^T)^{-1}}{\partial \bar{a}}\otimes(A^{-1}-\hat{A}^{-1})\right]U_{\alpha^2}^{\alpha^2\times\alpha^2} \quad (6.26)$$

$$\frac{\partial^2 F_4}{\partial \bar{a}^2} = \left[I_{\alpha^2}\otimes U_{\alpha^3.\alpha}^{\alpha^3.\alpha\times\alpha^3.\alpha}\right]\left[\frac{\partial (A^T)^{-1}}{\partial \bar{a}}\otimes\frac{\partial A^{-1}}{\partial \bar{a}}\right]U_{\alpha^2}^{\alpha^2\times\alpha^2} + \left[I_{\alpha^2}\otimes I_{\alpha^2}\otimes U_{\alpha^2}^{\alpha^2\times\alpha^2}\right]\left[I_{\alpha^2}\otimes U_{\alpha^3.\alpha}^{\alpha^3.\alpha\times\alpha^3.\alpha}\right]\left[\frac{\partial A^{-1}}{\partial \bar{a}}\otimes\frac{\partial (A^T)^{-1}}{\partial \bar{a}}\right]U_{\alpha^2}^{\alpha^2\times\alpha^2}U_{\alpha^2}^{\alpha^2\times\alpha^2} \quad (6.27)$$

The Taylor series expansion for $F_4(A)$ about the point $\hat{\bar{a}}$ can be written as

$$F_4 = 0+0+\left[(\bar{a}-\hat{\bar{a}})^T\otimes(\bar{a}-\hat{\bar{a}})^T\otimes I_{\alpha^2}\right]\frac{\partial^2 F_4}{\partial \bar{a}^2}+\ldots \quad (6.28)$$

$$\begin{aligned}\Sigma_{A^{-1}} = E[F_4] = {} & \left[S_{\bar{a}}^T\otimes I_{\alpha^2}\right]\{\left(I_{\alpha^2}\otimes U_{\alpha^3.\alpha}^{\alpha^3.\alpha\times\alpha^3.\alpha}\right)\\ & \left[\left(I_{\alpha^2}\otimes[A^{-1}]^T\right)V_{\alpha^2}^{\alpha^3\times\alpha}(A^T)^{-1}\otimes\left(I_{\alpha^2}\otimes A^{-1}\right)\bar{V}_{\alpha^2}^{\alpha^3\times\alpha}A^{-1}\right]\\ & U_{\alpha^2}^{\alpha^2\times\alpha^2}+\left(I_{\alpha^2}\otimes I_{\alpha^2}\otimes U_{\alpha^2}^{\alpha^2\times\alpha^2}\right)\left(I_{\alpha^2}\otimes U_{\alpha^3.\alpha}^{\alpha^3.\alpha\times\alpha^3.\alpha}\right)\\ & \left[\left(I_{\alpha^2}\otimes A^{-1}\right)\bar{V}_{\alpha^2}^{\alpha^3\times\alpha}A^{-1}\otimes\left(I_{\alpha^2}\otimes[A^{-1}]^T\right)V_{\alpha^2}^{\alpha^3\times\alpha}(A^T)^{-1}\right]\\ & U_{\alpha^2}^{\alpha^2\times\alpha^2}U_{\alpha^2}^{\alpha^2\times\alpha^2}\}\end{aligned} \quad (6.29)$$

Therefore we can write the distribution for A^{-1} as

$$A^{-1} \approx N\left(\hat{A}^{-1},\Sigma_{A^{-1}}\right) \quad (6.30)$$

5. $[F_5(A)]_{\beta\times\alpha} = A^\dagger$

$$\frac{\partial F_5}{\partial \bar{a}} = \frac{\partial A^\dagger}{\partial \bar{a}} = -\left[I_{\alpha\beta}\otimes A^\dagger\right]\bar{V}_{\alpha\beta}^{\alpha^2\beta\times\beta}A^\dagger \quad (6.31)$$

The Taylor series expansion for $F_5(A)$ about the point $\hat{\bar{a}}$ can be written as

$$F_5 = \hat{A}^\dagger - \left[(\bar{a}-\hat{\bar{a}})^T \otimes I_\alpha\right]\left[I_{\alpha\beta}\otimes A^{-1}\right]\bar{V}_{\alpha\beta}^{\alpha^2\beta\times\beta}A^\dagger + \ldots \tag{6.32}$$

$$E[A^\dagger] = \hat{A}^\dagger \tag{6.33}$$

6. $[F_6(A)]_{\alpha\beta\times\alpha\beta} = (A^\dagger - \hat{A}^\dagger)\otimes(A^\dagger-\hat{A}^\dagger)^T$

$$\frac{\partial F_6}{\partial\bar{a}} = \frac{\partial A^\dagger}{\partial\bar{a}}\otimes(A^\dagger-\hat{A}^\dagger)^T + \left[I_{\alpha\beta}\otimes U_{\beta\alpha}^{\beta\alpha\times\beta\alpha}\right]\left[\frac{\partial(A^\dagger)^T}{\partial\bar{a}}\otimes(A^\dagger-\hat{A}^\dagger)\right]U_{\alpha\beta}^{\alpha\beta\times\alpha\beta} \tag{6.34}$$

$$\begin{aligned}\frac{\partial^2 F_6}{\partial\bar{a}^2} = {} & \left[I_{\alpha\beta}\otimes U_{\alpha\beta^2.\alpha}^{\alpha\beta^2.\alpha\times\alpha\beta^2.\alpha}\right]\left[\frac{\partial(A^\dagger)^T}{\partial\bar{a}}\otimes\frac{\partial A^\dagger}{\partial\bar{a}}\right]U_{\alpha\beta}^{\alpha\beta\times\alpha\beta} \\ & + \left[I_{\alpha\beta}\otimes I_{\alpha\beta}\otimes U_{\beta\alpha}^{\beta\alpha\times\beta\alpha}\right]\left[I_{\alpha m}\otimes U_{\alpha^2\beta.\beta}^{\alpha^2\beta.\beta\times\alpha^2\beta.\beta}\right] \\ & \left[\frac{\partial A^\dagger}{\partial\bar{a}}\otimes\frac{\partial(A^\dagger)^T}{\partial\bar{a}}\right]U_{\beta\alpha}^{\beta\alpha\times\beta\alpha}U_{\alpha\beta}^{\alpha\beta\times\alpha\beta}\end{aligned} \tag{6.35}$$

Moreover,

$$\frac{\partial A^\dagger}{\partial\bar{a}} = -\left[I_{\alpha\beta}\otimes A^\dagger\right]\bar{V}_{\alpha\beta}^{\alpha^2\beta\times\beta}A^\dagger \tag{6.36}$$

The Taylor series expansion for $F_6(A)$ about the point $\hat{\bar{a}}$ can be written as

$$F_6 = 0+0+\left[(\bar{a}-\hat{\bar{a}})^T\otimes(\bar{a}-\hat{\bar{a}})^T\otimes I_{\alpha\beta}\right]\frac{\partial^2 F_6}{\partial\bar{a}^2} \tag{6.37}$$

$$\begin{aligned}\Sigma_{A^\dagger} = E[F_6] = {} & \left[S_{\bar{a}}^T\otimes I_{\alpha\beta}\right]\{\left[I_{\alpha\beta}\otimes U_{\alpha\beta^2.\alpha}^{\alpha\beta^2.\alpha\times\alpha\beta^2.\alpha}\right] \\ & \left[\left(I_{\alpha\beta}\otimes(A^T)^\dagger\right)V_{\alpha\beta}^{\alpha\beta^2\times\alpha}(A^T)^\dagger\otimes\left(I_{\alpha\beta}\otimes A^\dagger\right)\bar{V}_{\alpha\beta}^{\alpha^2\beta\times\beta}A^\dagger\right] \\ & U_{\alpha\beta}^{\alpha\beta\times\alpha\beta} + \left[I_{\alpha\beta}\otimes I_{\alpha\beta}\otimes U_{\beta\alpha}^{\beta\alpha\times\beta\alpha}\right]\left[I_{\alpha\beta}\otimes U_{\alpha^2\beta.\beta}^{\alpha^2\beta.\beta\times\alpha^2\beta.\beta}\right] \\ & \left[\left(I_{\alpha\beta}\otimes A^\dagger\right)\bar{V}_{\alpha\beta}^{\alpha^2\beta\times\beta}A^\dagger\otimes\left(I_{\alpha\beta}\otimes(A^T)^\dagger\right)V_{\alpha\beta}^{\alpha\beta^2\times\alpha}(A^T)^\dagger\right] \\ & U_{\beta\alpha}^{\beta\alpha\times\beta\alpha}U_{\alpha\beta}^{\alpha\beta\times\alpha\beta}\}\end{aligned} \tag{6.38}$$

Therefore we can write the distribution for $A^\dagger$ as

$$A^\dagger \approx N\left(\hat{A}^\dagger,\Sigma_{A^\dagger}\right) \tag{6.39}$$

7. $[F_7(A,B)]_{\alpha\times\rho} = A_{\alpha\times\beta}B_{\beta\times\rho}$

Note: We assume $\beta=\sigma$ in functions F_7 and F_8 for dimensional compatibility.

$$\frac{\partial F_7}{\partial\bar{a}} = \frac{\partial A}{\partial\bar{a}}B; \tag{6.40}$$

$$\frac{\partial F_7}{\partial \bar{b}} = [I_{\beta\rho} \otimes A] \frac{\partial B}{\partial \bar{b}}; \tag{6.41}$$

$$\frac{\partial^2 F_7}{\partial \bar{b}\ \partial \bar{a}} = \left[I_{\beta\rho} \otimes \frac{\partial A}{\partial \bar{a}}\right] \frac{\partial B}{\partial \bar{b}}; \tag{6.42}$$

$$\frac{\partial^2 F_7}{\partial \bar{a}\ \partial \bar{b}} = \tag{6.43}$$

The Taylor series expansion for $F_7(A, B)$ about the points $\{\hat{\bar{a}}, \hat{\bar{b}}\}$ can be written as

$$\begin{aligned} F_7 &= \hat{A}\hat{B} + \left[(\bar{a} - \hat{\bar{a}})^T \otimes I_\alpha\right] \frac{\partial F_7}{\partial \bar{a}} + \left[(\bar{b} - \hat{\bar{b}})^T \otimes I_\alpha\right] \frac{\partial F_7}{\partial \bar{b}} \\ &\quad + \frac{1}{2}\left[(\bar{a} - \hat{\bar{a}})^T \otimes (\bar{b} - \hat{\bar{b}})^T \otimes I_\alpha\right] \frac{\partial^2 F_7}{\partial \bar{a}\ \partial \bar{b}} \\ &\quad + \frac{1}{2}\left[(\bar{b} - \hat{\bar{b}})^T \otimes (\bar{a} - \hat{\bar{a}})^T \otimes I_\alpha\right] \frac{\partial^2 F_7}{\partial \bar{b}\ \partial \bar{a}} + \ldots \\ E[AB] &= \hat{A}\hat{B} \end{aligned} \tag{6.44}$$

8. $[F_8(A, B)]_{\alpha\rho \times \alpha\rho} = (AB - \hat{A}\hat{B}) \otimes (AB - \hat{A}\hat{B})^T$

$$\begin{aligned} \frac{\partial F_8}{\partial \bar{a}} &= \left[\frac{\partial A}{\partial \bar{a}} B \otimes (AB - \hat{A}\hat{B})^T\right] \\ &\quad + \left[I_{\alpha\beta} \otimes U_{\alpha\rho}^{\alpha\rho \times \alpha\rho}\right] \left[(I_{\alpha\beta} \otimes B^T) \frac{\partial A^T}{\partial \bar{a}} \otimes (AB - \hat{A}\hat{B})\right] U_{\rho\alpha}^{\rho\alpha \times \rho\alpha} \end{aligned} \tag{6.45}$$

$$\begin{aligned} \frac{\partial F_8}{\partial \bar{b}} &= \left[(I_{\beta\rho} \otimes A) \frac{\partial B}{\partial \bar{b}} \otimes (AB - \hat{A}\hat{B})^T\right] \\ &\quad + \left[I_{\beta\rho} \otimes U_{\alpha\rho}^{\alpha\rho \times \alpha\rho}\right] \left[\frac{\partial B^T}{\partial \bar{b}} A^T \otimes (AB - \hat{A}\hat{B})\right] U_{\rho\alpha}^{\rho\alpha \times \rho\alpha} \end{aligned} \tag{6.46}$$

$$\begin{aligned} \frac{\partial^2 F_8}{\partial \bar{a}^2} &= \left[I_{\alpha\beta} \otimes U_{\alpha^2\beta.\rho}^{\alpha^2\beta.\rho \times \alpha^2\beta.\rho}\right] \left[(I_{\alpha\beta} \otimes B^T) \frac{\partial A^T}{\partial \bar{a}} \otimes \frac{\partial A}{\partial \bar{a}} B\right] U_{\rho\alpha}^{\rho\alpha \times \rho\alpha} \\ &\quad + \left[I_{\alpha\beta} \otimes I_{\alpha\beta} \otimes U_{\alpha\rho}^{\alpha\rho \times \alpha\rho}\right] \{\left[I_{\alpha\beta} \otimes U_{\alpha\beta\alpha.\rho}^{\alpha\beta\rho.\alpha \times \alpha\beta\rho.\alpha}\right] \\ &\quad \left[\frac{\partial A}{\partial \bar{a}} B \otimes (I_{\alpha\beta} \otimes B^T) \frac{\partial A^T}{\partial \bar{a}}\right] U_{\alpha\rho}^{\alpha\rho \times \alpha\rho}\} U_{\rho\alpha}^{\rho\alpha \times \rho\alpha} \end{aligned} \tag{6.47}$$

$$\begin{aligned} \frac{\partial^2 F_8}{\partial \bar{b}^2} &= \left[I_{\beta l} \otimes U_{\alpha\beta\rho.\rho}^{\alpha\beta\rho.\rho \times \alpha\beta\rho.\rho}\right] \left[\frac{\partial B^T}{\partial \bar{b}} A^T \otimes (I_{\beta\rho} \otimes A) \frac{\partial B}{\partial \bar{b}}\right] U_{\rho\alpha}^{\rho\alpha \times \rho\alpha} \\ &\quad + \left[I_{\beta\rho} \otimes I_{\beta\rho} \otimes U_{\alpha\rho}^{\alpha\rho \times \alpha\rho}\right] \{\left[I_{\beta\rho} \otimes U_{\beta l^2.\alpha}^{\beta\rho^2.\alpha \times \beta\rho^2.\alpha}\right] \\ &\quad \left[(I_{\beta\rho} \otimes A) \frac{\partial B}{\partial \bar{b}} \otimes \frac{\partial B^T}{\partial \bar{b}} A^T\right] U_{\alpha\rho}^{\alpha\rho \times \alpha\rho}\} U_{\rho\alpha}^{\rho\alpha \times \rho\alpha} \end{aligned} \tag{6.48}$$

Note that we do not have to derive $\left\{\frac{\partial^2 F_8}{\partial \bar{b}\partial \bar{a}}, \frac{\partial^2 F_8}{\partial \bar{a}\partial \bar{b}}\right\}$ because the terms involving $\left\{\left[(\bar{a}-\hat{\bar{a}})^T \otimes (\bar{b}-\hat{\bar{b}})^T \otimes I_{\alpha\rho}\right], \left[(\bar{b}-\hat{\bar{b}})^T \otimes (\bar{a}-\hat{\bar{a}})^T \otimes I_{\alpha\rho}\right]\right\}$ are zeros (since $\bar{a}$ is assumed uncorrelated with $\bar{b}$).

The Taylor series expansion for $F_8(A,B)$ about the points $\{\hat{\bar{a}}, \hat{\bar{b}}\}$ can be written as

$$\begin{aligned} F_8 &= 0+0+0+\left[(\bar{a}-\hat{\bar{a}})^T \otimes (\bar{a}-\hat{\bar{a}})^T \otimes I_{\alpha\rho}\right]\frac{\partial^2 F_8}{\partial \bar{a}^2} \\ &\quad + \left[(\bar{b}-\hat{\bar{b}})^T \otimes (\bar{b}-\hat{\bar{b}})^T \otimes I_{\alpha\rho}\right]\frac{\partial^2 F_8}{\partial \bar{b}^2} + 0+0+\ldots \qquad (6.49) \\ \Sigma_{AB} &= \left[S_{\bar{a}}^T \otimes I_{\alpha\rho}\right]\left\{\left[I_{\alpha\beta} \otimes U_{\alpha^2\beta.\rho}^{\alpha^2\beta.\rho\times\alpha^2\beta.\rho}\right]\left[(I_{\alpha\beta}\otimes B^T)V_{\alpha\beta}^{\alpha\beta^2\times\alpha} \otimes \bar{V}_{\alpha\beta}^{\alpha^2\beta\times\beta}B\right]U_{\rho\alpha}^{\rho\alpha\times\rho\alpha}\right. \\ &\quad + \left[I_{\alpha\beta}\otimes I_{\alpha\beta}\otimes U_{\alpha\rho}^{\alpha\rho\times\alpha\rho}\right]\left[I_{\alpha\beta}\otimes U_{\alpha\beta\rho.\alpha}^{\alpha\beta\rho.\alpha\times\alpha\beta\rho.\alpha}\right] \\ &\quad \left.\left[\bar{V}_{\alpha\beta}^{\alpha^2\beta\times\beta}B\otimes(I_{\alpha\beta}\otimes B^T)V_{\alpha\beta}^{\alpha\beta^2\times\alpha}\right]U_{\alpha\rho}^{\alpha\rho\times\alpha\rho}U_{\rho\alpha}^{\rho\alpha\times\rho\alpha}\right\} \\ &\quad + \left[S_{\bar{b}}^T \otimes I_{\alpha\rho}\right]\left\{\left[I_{\beta\rho}\otimes U_{\alpha\beta\rho.\rho}^{\alpha\beta\rho.\rho\times\alpha\beta\rho.\rho}\right]\left[V_{\beta\rho}^{\beta\rho^2\times\beta}A^T\otimes(I_{\beta\rho}\otimes A)\bar{V}_{\beta\rho}^{\beta^2\rho\times\rho}\right]U_{\rho\alpha}^{\rho\alpha\times\rho\alpha}\right. \\ &\quad + \left[I_{\beta\rho}\otimes I_{\beta\rho}\otimes U_{\alpha\rho}^{\alpha\rho\times\alpha\rho}\right]\left[I_{\beta\rho}\otimes U_{\beta\rho^2.\alpha}^{\beta\rho^2.\alpha\times\beta\rho^2.\alpha}\right] \\ &\quad \left.\left[(I_{\beta\rho}\otimes A)\bar{V}_{\beta\rho}^{\beta^2\rho\times\rho}\otimes V_{\beta\rho}^{\beta\rho^2\times\beta}A^T\right]U_{\alpha\rho}^{\alpha\rho\times\alpha\rho}U_{\rho\alpha}^{\rho\alpha\times\rho\alpha}\right\} \qquad (6.50) \end{aligned}$$

Therefore we can write the distribution for AB as

$$AB \approx N\left(\hat{A}\hat{B}, \Sigma_{AB}\right) \qquad (6.51)$$

The above results will be used in the derivation of the statistical properties of the subspace matrices and subsequently the confidence limits for the LQG-benchmark curve.

6.5 Multivariate multiple regression

Most of the presentation in this section is taken from [55] with some matrix dimensional modifications to suit this chapter. References at the end of the thesis can guide readers who may be interested in the detailed derivations. A multivariate multiple regression model can be expressed as

$$\underset{(a\times b)}{X} = \underset{(a\times c)}{\beta}\,\underset{(c\times b)}{Z} + \underset{(a\times b)}{\Xi} \qquad (6.52)$$

for $\Xi_{(i)} = \Xi(i,:) \quad i = 1, 2, ..., a$ with

$$E[\Xi_{(i)}] = 0; \qquad Cov(\Xi_{(i)}, \Xi_{(k)}) = \sigma_{ik} I \qquad i, k = 1, 2, ..., a$$

where X contains b trials of a responses, Z contains b trials of c predictor variables and Ξ contains the error terms. The a observations on the j-th trial have the generalized variance matrix $\Sigma = \{\sigma_{ik}\}$, but observations from different trials are uncorrelated. Here β and σ_{ik} are unknown parameters.

Given the outcomes X and the predictor variables Z with full row rank, we can determine the least squares estimates $\hat{\beta}_{(i)}$ exclusively from the observations, $X_{(i)}$, on the i-th response.

$$\hat{\beta}_{(i)} = X_{(i)} Z^T (ZZ^T)^{-1} \tag{6.53}$$

Collecting the univariate least squares estimates, we obtain

$$\hat{\beta} = \begin{bmatrix} \hat{\beta}_{(1)} & \hat{\beta}_{(2)} & ... & \hat{\beta}_{(b)} \end{bmatrix} = \begin{bmatrix} X_{(1)} & X_{(2)} & ... & X_{(b)} \end{bmatrix} Z^T (ZZ^T)^{-1} \tag{6.54}$$

Note that the least squares estimates, $\hat{\beta}_{(i)} = X_{(i)} Z^T (ZZ^T)^{-1}$, are computed individually for each response variable. However, the model requires that the same predictor variables be used for all responses. Using the least squares estimate, $\hat{\beta}$, we can form the residuals: $\hat{\Xi} = X - \hat{\beta} Z$. The estimated error sum of squares and cross-products matrix is

$$\hat{\Sigma}_{\Xi} = \frac{\hat{\Xi}^T \hat{\Xi}}{b} = \frac{(X - \hat{\beta} Z)(X - \hat{\beta} Z)^T}{b} = \begin{bmatrix} \hat{\sigma}_{11} & ... & \hat{\sigma}_{1b} \\ ... & ... & ... \\ \hat{\sigma}_{b1} & ... & \hat{\sigma}_{bb} \end{bmatrix} \tag{6.55}$$

Let the multivariate multiple regression model in equation (6.52) hold with full rank$(Z) = a$, $b \geq c + a$ and let the errors, Ξ, have a normal distribution. Then

$$\hat{\beta} = X Z^T (ZZ^T)^{-1} \tag{6.56}$$

is the MLE (maximum likelihood estimator) of β.

Consider $\hat{\beta}_i = (X_{(i)} - \Xi_i) Z^T (ZZ^T)^{-1}$ and $\beta_i = X_{(i)} Z^T (ZZ^T)^{-1}$. Therefore

$$\beta_i - \hat{\beta}_i = \Xi_i Z^T (ZZ^T)^{-1} \tag{6.57}$$

$$Cov(\hat{\beta}_i, \hat{\beta}_k) = E[(\beta_i - \hat{\beta}_i)^T(\beta_k - \hat{\beta}_k)] \quad (6.58)$$

$$= (ZZ^T)^{-1}ZE[\Xi_i^T\Xi_k]Z^T(ZZ^T)^{-1} = \sigma_{ik}(ZZ^T)^{-1} \quad (6.59)$$

Hence $\hat{\beta}$ has a normal distribution with $E[\hat{\beta}]$ and covariance

$$\hat{\Sigma}_\beta = \hat{\Sigma_\Xi} \otimes (ZZ^T)^\dagger \quad (6.60)$$

Also, $\hat{\beta}$ is independent of $\hat{\Sigma}$, the MLE of the positive definite matrix Σ. For large samples, $\hat{\beta}$ and $\hat{\Sigma}$, have the smallest possible variances.

6.6 Statistical properties for subspace matrices

Calculation of the LQG-benchmark variances of the process input and output involves the estimation of the subspace matrices L_u and L_e using the process data. The subspace matrices may be identified using the process open-loop data (see chapter 2) or closed-loop data (see chapter 3).

6.6.1 Subspace matrices identification from the open-loop process data

The subspace matrices L_u and L_e can be estimated from the open-loop data using the following linear expression:

$$\underset{mN \times j}{Y_f} = \underset{mN \times (m+l)N}{L_w}\ \underset{(m+l)N \times j}{W_p} + \underset{mN \times lN}{L_u}\ \underset{lN \times j}{U_f} + \underset{mN \times mN}{L_e}\ \underset{mN \times j}{E_f} \quad (6.61)$$

Note that we have no data to build the matrix E_f. Hence the data Hankel matrix, E_f is formed from the estimated innovations sequence. The estimated noise sequence is the one step ahead predictions, $P_f(1:m,:)$ where $P_f = Y_f - L_wW_p - L_uU_f$. Hence the subspace matrices $\{L_w, L_u \text{ and } L_e\}$ are estimated in two steps:

(i) First the parametric matrices $\{L_w, L_u\}$ are estimated as

$$\begin{bmatrix} \hat{L}_w & \hat{L}_u \end{bmatrix} = Y_f \begin{bmatrix} W_p \\ U_f \end{bmatrix}^\dagger \quad (6.62)$$

Then

(*ii*) the data Hankel matrix $\hat{E}_f$ is built (see equations (2.4)-(2.7) on building data Hankel matrices) from the first block row of the residuals, P_f. L_e is estimated as

$$\hat{L}_e = P_f \hat{E}_f^{\dagger} \tag{6.63}$$

The residuals can be obtained as

$$\Xi = Y_f - \hat{L}_w W_p - \hat{L}_u U_f - \hat{L}_e \hat{E}_f \tag{6.64}$$

The covariance matrix of the residuals is estimated as

$$\underset{mN \times mN}{\hat{\Sigma}_{\Xi}} = \frac{\hat{\Xi}\hat{\Xi}^T}{j} = \begin{bmatrix} \hat{\sigma}_{1,1} & \cdots & \hat{\sigma}_{1,mN} \\ \cdots & \cdots & \cdots \\ \hat{\sigma}_{mN,1} & \cdots & \hat{\sigma}_{mN,mN} \end{bmatrix} \tag{6.65}$$

The covariance matrices for the subspace matrices $\{\Sigma_{L_u}, \Sigma_{L_e}\}$ are estimated as follows

$$\Sigma_{\beta^{OL}} = \hat{\Sigma}_{\Xi} \otimes \left(\begin{bmatrix} W_p \\ U_f \end{bmatrix} \begin{bmatrix} W_p \\ U_f \end{bmatrix}^T \right)^{-1} \tag{6.66}$$

$$\Sigma_{\hat{L}_u} = \Sigma_{\beta}[(m^2+ml)N^2+1 : (m^2+2ml)N^2, \; (m^2+ml)N^2+1 : (m^2+2ml)N^2] \tag{6.67}$$

$$\Sigma_{\hat{L}_e} = \hat{\Sigma}_{\Xi} \otimes (\hat{E}_f \hat{E}_f^T)^{-1} \tag{6.68}$$

6.6.2 Subspace matrices identification from the closed-loop process data

Estimation of the required closed loop subspace matrices, L_{yr}^{Cl}, L_{ur}^{CL} and L_{ue}^{CL}, using the least squares estimation method. Estimation of the closed loop subspace matrices from the closed loop data is an open loop identification problem:

$$\underset{(m+l)N \times j}{\begin{bmatrix} Y_f \\ U_f \end{bmatrix}} = \underset{(m+l)N \times (2m+l)N}{\begin{bmatrix} L_y^{CL} \\ L_u^{CL} \end{bmatrix}} \underset{(2m+l)N \times j}{M_p}$$

$$+ \underset{(m+l)N \times mN}{\begin{bmatrix} L_{yr}^{CL} \\ L_{ur}^{CL} \end{bmatrix}} \underset{mN \times j}{R_f} + \underset{(m+l)N \times mN}{\begin{bmatrix} L_{ye}^{CL} \\ L_{ue}^{CL} \end{bmatrix}} \underset{mN \times j}{E_f} \quad (6.69)$$

Similar to the previous section the subspace matrices $\{L_y^{CL}, L_u^{CL}, L_{yr}^{CL}, L_{ur}^{CL}, L_{ye}^{CL}$ and $L_{ue}^{CL}\}$ are estimated in two steps:

(i) First the parametric matrices $\{L_y^{CL}, L_u^{CL}, L_{yr}^{CL}, L_{ur}^{CL}$ are estimated as

$$\begin{bmatrix} \hat{L}_y^{CL} & \hat{L}_{yr}^{CL} \\ \hat{L}_u^{CL} & \hat{L}_{ur}^{CL} \end{bmatrix} = \begin{bmatrix} Y_f \\ U_f \end{bmatrix} \begin{bmatrix} M_p \\ R_f \end{bmatrix}^{\dagger} \quad (6.70)$$

Then

(ii) the data Hankel matrix $\hat{E}_f$ is built (see equations (2.4)-(2.7)) from the first block row of the residuals, $P_f^{CL} = U_f - L_u^{CL} M_p - L_{ur}^{CL} R_f$. L_{ue}^{CL} is estimated as

$$\hat{L}_{ue}^{CL} = P_f^{CL} \hat{E}_f^{\dagger} \quad (6.71)$$

We can obtain the residuals as

$$\Xi = \begin{bmatrix} Y_f - \hat{L}_y^{CL} M_p - \hat{L}_{yr}^{CL} R_f - \hat{L}_{ye}^{CL} \hat{E}_f \\ Y_f - \hat{L}_u^{CL} M_p - \hat{L}_{ur}^{CL} R_f - \hat{L}_{ye}^{CL} \hat{E}_f \end{bmatrix} \quad (6.72)$$

The covariance matrix of the residuals is estimated as

$$\underset{(m+l)N \times (m+l)N}{\hat{\Sigma}_{\Xi}} = \frac{\hat{\Xi}\hat{\Xi}^T}{j} = \begin{bmatrix} \hat{\sigma}_{1,1} & \cdots & \hat{\sigma}_{1,(m+l)N} \\ \cdots & \cdots & \cdots \\ \hat{\sigma}_{(m+l)N,1} & \cdots & \hat{\sigma}_{(m+l)N,(m+l)N} \end{bmatrix} \quad (6.73)$$

The covariance matrices for the closed loop subspace matrices $\{\Sigma_{L_{yr}^{CL}}, \Sigma_{L_{ur}^{CL}}, \Sigma_{L_{ue}^{CL}}\}$ are then estimated as follows

$$\Sigma_{\beta^{CL}} = \hat{\Sigma}_{\Xi} \otimes \left(\begin{bmatrix} M_p \\ R_f \end{bmatrix} \begin{bmatrix} M_p \\ R_f \end{bmatrix}^T \right)^{-1} \quad (6.74)$$

$$\Sigma_{L_{yr}^{CL}} = \Sigma_{\beta}[(2m^2 + l^2 + 3ml)N^2 + 1 : (3m^2 + l^2 + 3ml)N^2, \\ (2m^2 + l^2 + 3ml)N^2 + 1 : (3m^2 + l^2 + 3ml)N^2] \quad (6.75)$$

$$\Sigma_{\hat{L}_{ur}^{CL}} = \Sigma_\beta[(3m^2+l^2+3ml)N^2+1:(3m^2+l^2+4ml)N^2, (3m^2+l^2+3ml)N^2+1:(3m^2+l^2+4ml)N^2] \quad (6.76)$$

$$\Sigma_{\hat{L}_{ue}^{CL}} = \hat{\Sigma}_\Xi \otimes (\hat{E}_f \hat{E}_f^T)^{-1} \quad (6.77)$$

The open loop subspace matrices, $\hat{L}_u(mN \times lN)$ and $\hat{L}_e(mN \times mN)$, are estimated from the closed loop subspace matrices $\hat{L}_{yr}^{CL}(mN \times mN)$, $\hat{L}_{ur}^{CL}(lN \times mN)$ and $\hat{L}_{ue}^{CL}(lN \times mN)$ as follows

$$\hat{L}_u = \hat{L}_{yr}^{CL}(\hat{L}_{ur}^{CL})^\dagger \quad (6.78)$$

$$\hat{L}_e = -(\hat{L}_{ur}^{CL})^\dagger \hat{L}_{ue}^{CL} \quad (6.79)$$

The covariance expressions for the open loop subspace matrices $\hat{L}_u$ and $\hat{L}_e$, are then derived as follows

$$(L_{ur}^{CL})^\dagger \approx N\left((\hat{L}_{ur}^{CL})^\dagger, \Sigma_{(L_{ur}^{CL})^\dagger}\right) \quad (6.80)$$

$$\begin{aligned}\Sigma_{(L_{ur}^{CL})^\dagger} = {} & \left[S_{L_{ur}^{CL}}^T \otimes I_{mlN^2}\right]\{\left[I_{mlN^2} \otimes U_{m^2l^2N^4}^{m^2l^2N^4 \times m^2l^2N^4}\right] \\ & \left[\left(I_{mlN^2} \otimes ([\hat{L}_{ur}^{CL}]^T)^\dagger\right) V_{mlN^2}^{m^2lN^3 \times lN}([\hat{L}_{ur}^{CL}]^T)^\dagger \otimes \left(I_{mlN^2} \otimes (\hat{L}_{ur}^{CL})^\dagger\right) \bar{V}_{mlN^2}^{ml^2N^3 \times mN}(\hat{L}_{ur}^{CL})^\dagger\right] \\ & U_{mlN^2}^{mlN^2 \times mlN^2} + \left[I_{mlN^2} \otimes I_{mlN^2} \otimes U_{mlN^2}^{mlN^2 \times mlN^2}\right]\left[I_{mlN^2} \otimes U_{m^2l^2N^4}^{m^2l^2N^4 \times m^2l^2N^4}\right] \\ & \left[\left(I_{mlN^2} \otimes (\hat{L}_{ur}^{CL})^\dagger\right) \bar{V}_{mlN^2}^{ml^2N^3 \times mN}(\hat{L}_{ur}^{CL})^\dagger \otimes \left(I_{mlN^2} \otimes ([\hat{L}_{ur}^{CL}]^T)^\dagger\right) V_{mlN^2}^{m^2lN^3 \times lN}([\hat{L}_{ur}^{CL}]^T)^\dagger\right] \\ & U_{mlN^2}^{mlN^2 \times mlN^2} U_{mlN^2}^{mlN^2 \times mlN^2}\}\end{aligned} \quad (6.81)$$

$$L_u = L_{yr}^{CL}(L_{ur}^{CL})^\dagger \approx N\left(\hat{L}_u, \Sigma_{L_u}\right) \quad (6.82)$$

$$\begin{aligned}\Sigma_{L_u} = {} & \left[S_{L_{yr}^{CL}}^T \otimes I_{mlN^2}\right]\{\left[I_{m^2N^2} \otimes U_{m^3lN^4}^{m^3lN^4 \times m^3lN^4}\right] \\ & \left[(I_{m^2N^2} \otimes P_1^T)V_{m^2N^2}^{m^3N^3 \times mN} \otimes \bar{V}_{m^2N^2}^{m^3N^3 \times mN} P_1\right] U_{lmN^2}^{lmN^2 \times lmN^2} \\ & + \left[I_{m^2N^2} \otimes I_{m^2N^2} \otimes U_{lmN^2}^{lmN^2 \times lmN^2}\right]\left[I_{m^2N^2} \otimes U_{lm^3N^4}^{lm^3N^4 \times lm^3N^4}\right] \\ & \left[\bar{V}_{m^2N^2}^{m^3N^3 \times mN} P_1 \otimes (I_{m^2N^2} \otimes P_1^T)V_{m^2N^2}^{m^3N^3 \times mN}\right] U_{lmN^2}^{mlN^2 \times mlN^2} U_{mlN^2}^{mlN^2 \times mlN^2}\} \\ & + \left[S_{(L_{ur}^{CL})^\dagger}^T \otimes I_{lmN^2}\right]\{\left[I_{lmN^2} \otimes U_{m^2l^2N^4}^{m^2l^2N^4 \times m^2l^2N^4}\right] \\ & \left[V_{mlN^2}^{ml^2N^3 \times mN}(L_c^O)^T \otimes (I_{mlN^2} \otimes L_c^O)\bar{V}_{mlN^2}^{m^2lN^3 \times lN}\right] U_{mlN^2}^{mlN^2 \times mlN^2} \\ & + \left[I_{mlN^2} \otimes I_{mlN^2} \otimes U_{mlN^2}^{mlN^2 \times mlN^2}\right]\left[I_{mlN^2} \otimes U_{m^2l^2N^4}^{m^2l^2N^4 \times m^2l^2N^4}\right] \\ & \left[(I_{mlN^2} \otimes L_c^O)\bar{V}_{mlN^2}^{m^2lN^3 \times lN} \otimes V_{mlN^2}^{ml^2N^3 \times mN}(L_c^O)^T\right] U_{mlN^2}^{mlN^2 \times mlN^2} U_{mlN^2}^{mlN^2 \times mlN^2}\}\end{aligned} \quad (6.83)$$

$$L_e = -(L_{ur}^{CL})^\dagger L_{ue}^{CL} \approx N\left(\hat{L}_e, \Sigma_{L_e}\right) \quad (6.84)$$

$$
\begin{aligned}
\Sigma_{L_e} = & \left[S^T_{(L^{CL}_{ur})^\dagger} \otimes I_{nl}\right] \{ \left[I_{mlN^2} \otimes U^{m^3lN^4 \times m^3lN^4}_{m^3lN^4}\right] \\
& \left[(I_{mlN^2} \otimes (L^I_e)^T) V^{ml^2N^3 \times mN}_{mlN^2} \otimes \bar{V}^{m^2lN^3 \times lN}_{mlN^2} L^{CL}_{ue}\right] U^{m^2N^2 \times m^2N^2}_{m^2N^2} \\
& + \left[I_{mlN^2} \otimes I_{mlN^2} \otimes U^{m^2N^2 \times m^2N^2}_{m^2N^2}\right] \left[I_{mlN^2} \otimes U^{m^3lN^4 \times m^3lN^4}_{m^3lN^4}\right] \\
& \left[\bar{V}^{m^2lN^3 \times lN}_{mlN^2} L^{CL}_{ue} \otimes (I_{mlN^2} \otimes (L^{CL}_{ue})^T) V^{ml^2N^3 \times mN}_{mlN^2}\right] U^{m^2N^2 \times m^2N^2}_{m^2N^2} U^{m^2N^2 \times m^2N^2}_{m^2N^2} \} \\
& + \left[S^T_{L^{CL}_{ue}} \otimes I_{m^2N^2}\right] \{ \left[I_{lmN^2} \otimes U^{m^3lN^4 \times m^3lN^4}_{m^3lN^4}\right] \\
& \left[V^{ml^2 \times m}_{ml}((L^{CL}_{ur})^\dagger)^T \otimes (I_{lmN^2} \otimes (L^{CL}_{ur})^\dagger) \bar{V}^{l^2mN^3 \times mN}_{lmN^2}\right] U^{m^2N^2 \times m^2N^2}_{m^2N^2} \\
& + \left[I_{lmN^2} \otimes I_{mlN^2} \otimes U^{m^2N^2 \times m^2N^2}_{m^2N^2}\right] \left[I_{mlN^2} \otimes U^{m^3lN^4 \times m^3lN^4}_{m^3lN^4}\right] \\
& \left[(I_{mlN^2} \otimes (L^{CL}_{ur})^\dagger) \bar{V}^{l^2mN^3 \times mN}_{mlN^2} \otimes V^{m^2lN^3 \times lN}_{mlN^2}((L^{CL}_{ur})^\dagger)^T\right] \\
& U^{m^2N^2 \times m^2N^2}_{m^2N^2} U^{m^2N^2 \times m^2N^2}_{m^2N^2} \} \qquad (6.85)
\end{aligned}
$$

We can also write

$$\hat{l}_e = \hat{L}_e(:, 1:m) \qquad (6.86)$$

Therefore

$$\Sigma_{l_e} = \Sigma_{\hat{L}_e}(1:m^2N, 1:m^2N) \qquad (6.87)$$

6.7 Statistical properties for the estimated LQG benchmark coefficients

The estimated matrices in the calculation of the LQG-benchmark variances can be written as

$$\hat{K} = (\lambda I + \hat{L}_u^T \hat{L}_u)^{-1} \hat{L}_u^T \qquad (6.88)$$

$$LQG_y = (I - \hat{L}_u \hat{K}) \hat{l}_e \qquad (6.89)$$

$$LQG_u = -\hat{K} \hat{l}_e \qquad (6.90)$$

Using the expressions derived in section 6.4, we can write the distributions for the estimated open loop subspace matrices and the LQG-benchmark variances as follows:

$$L_u^T \approx N\left(\hat{L}_u^T, \Sigma_{L_u}\right) \qquad (6.91)$$

$$F_1 = L_u^T L_u = \approx N\left(\hat{L}_u^T \hat{L}_u, \Sigma_{F_1}\right) \tag{6.92}$$

$$\begin{aligned}
\Sigma_{F_1} = & \left[S_{L_u}^T \otimes I_{l^2N^2}\right]\left\{\left[I_{mlN^2} \otimes U_{ml^3N^4}^{ml^3N^4 \times ml^3N^4}\right]\right. \\
& \left[(I_{mlN^2} \otimes L_u^T) V_{mlN^2}^{m^2lN^3 \times lN} \otimes \bar{V}_{mlN^2}^{l^2mN^3 \times mN} L_u\right] U_{l^2N^2}^{l^2N^2 \times l^2N^2} \\
& + \left[I_{mlN^2} \otimes I_{mlN^2} \otimes U_{l^2N^2}^{l^2N^2 \times l^2N^2}\right]\left[I_{mlN^2} \otimes U_{l^3mN^4}^{l^3mN^4 \times l^3mN^4}\right] \\
& \left.\left[\bar{V}_{mlN^2}^{l^2mN^3 \times m} L_u \otimes (I_{mlN^2} \otimes L_u^T) V_{mlN^2}^{m^2lN^3 \times lN}\right] U_{l^2N^2}^{l^2N^2 \times l^2N^2} U_{l^2N^2}^{l^2N^2 \times l^2N^2}\right\} \\
& + \left[S_{L_u}^T \otimes I_{l^2N^2}\right]\left\{\left[I_{mlN^2} \otimes U_{ml^3N^4}^{ml^3N^4 \times ml^3N^4}\right]\right. \\
& \left[V_{mlN^2}^{ml^2N^3 \times mN} L_u \otimes (I_{mlN^2} \otimes L_u^T) \bar{V}_{mlN^2}^{m^2lN^3 \times lN}\right] U_{mlN^2}^{mlN^2 \times mlN^2} \\
& + \left[I_{mlN^2} \otimes I_{mlN^2} \otimes U_{l^2N^2}^{l^2N^2 \times l^2N^2}\right]\left[I_{mlN^2} \otimes U_{ml^3N^4}^{ml^3N^4 \times ml^3N^4}\right] \\
& \left.\left[(I_{mlN^2} \otimes L_u^T) \bar{V}_{mlN^2}^{m^2lN^3 \times lN} \otimes V_{mlN^2}^{ml^2N^3 \times mN} L_u\right] U_{l^2N^2}^{l^2N^2 \times l^2N^2} U_{l^2N^2}^{l^2N^2 \times l^2N^2}\right\}
\end{aligned} \tag{6.93}$$

$$F_2 = \lambda I + L_u^T L_u \approx N\left([\lambda I + \hat{L}_u^T \hat{L}_u], \Sigma_{F_1}\right) \tag{6.94}$$

$$F_3 = (\lambda I + L_u^T L_u)^{-1} \approx N\left([\lambda I + \hat{L}_u^T \hat{L}_u]^{-1}, \Sigma_{F_3}\right) \tag{6.95}$$

$$\begin{aligned}
\Sigma_{F_3} = & \left[S_{P_6}^T \otimes I_{n^2}\right]\left\{\left(I_{l^2N^2} \otimes U_{l^4N^4}^{l^4N^4 \times l^4N^4}\right)\right. \\
& \left[\left(I_{l^2N^2} \otimes [(\lambda I + L_u^T L_u)^{-1}]^T\right) V_{l^2N^2}^{l^3N^3 \times lN} [(\lambda I + L_u^T L_u)^T]^{-1}\right. \\
& \left.\otimes \left(I_{l^2N^2} \otimes (\lambda I + L_u^T L_u)^{-1}\right) \bar{V}_{l^2N^2}^{l^3N^3 \times lN} (\lambda I + L_u^T L_u)^{-1}] U_{l^2N^2}^{l^2N^2 \times l^2N^2}\right. \\
& + \left(I_{l^2N^2} \otimes I_{l^2N^2} \otimes U_{l^2N^2}^{l^2N^2 \times l^2N^2}\right)\left(I_{l^2N^2} \otimes U_{l^4N^4}^{l^4N^4 \times l^4N^4}\right) \\
& \left[\left(I_{l^2N^2} \otimes (\lambda I + L_u^T L_u)^{-1}\right) \bar{V}_{l^2N^2}^{l^3N^3 \times lN} (\lambda I + L_u^T L_u)^{-1}\right. \\
& \left.\otimes \left(I_{l^2N^2} \otimes [(\lambda I + L_u^T L_u)^{-1}]^T\right) V_{l^2N^2}^{l^3N^3 \times lN} [(\lambda I + L_u^T L_u)^T]^{-1}\right] \\
& \left.U_{l^2N^2}^{l^2N^2 \times l^2N^2} U_{l^2N^2}^{l^2N^2 \times l^2N^2}\right\}
\end{aligned} \tag{6.96}$$

$$K = (\lambda I + L_u^T L_u)^{-1} L_u^T \approx N\left(\hat{K}, \Sigma_K\right) \tag{6.97}$$

$$\begin{aligned}
\Sigma_K = & \left[S_{F_3}^T \otimes I_{mlN^2}\right]\left\{\left[I_{l^2N^2} \otimes U_{l^3mN^4}^{l^3mN^4 \times l^3mN^4}\right]\right. \\
& \left[(I_{l^2N^2} \otimes L_u) V_{l^2N^2}^{l^3N^3 \times lN} \otimes \bar{V}_{l^2N^2}^{l^3N^3 \times lN} L_u^T\right] U_{mlN^2}^{mlN^2 \times mlN^2} \\
& + \left[I_{l^2N^2} \otimes I_{l^2N^2} \otimes U_{lmN^2}^{lmN^2 \times lmN^2}\right]\left[I_{l^2N^2} \otimes U_{l^3mN^4}^{l^3mN^4 \times l^3mN^4}\right] \\
& \left.\left[\bar{V}_{l^2N^2}^{l^3N^3 \times lN} L_u^T \otimes (I_{l^2N^2} \otimes L_u) V_{l^2N^2}^{l^3N^3 \times lN}\right] U_{lmN^2}^{lmN^2 \times lmN^2} U_{lmN^2}^{lmN^2 \times lmN^2}\right\} \\
& + \left[S_{L_u}^T \otimes I_{lmN^2}\right]\left\{\left[I_{lmN^2} \otimes U_{l^2m^2N^4}^{l^2m^2N^4 \times l^2m^2N^4}\right]\right. \\
& \left[V_{mlN^2}^{m^2lN^3 \times lN} [(\lambda I + L_u^T L_u)^{-1}]^T \otimes (I_{mlN^2} \otimes (\lambda I + L_u^T L_u)^{-1}) \bar{V}_{mlN^2}^{l^2mN^3 \times mN}\right] \\
& U_{mlN^2}^{mlN^2 \times mlN^2} \\
& \left. + \left[I_{mlN^2} \otimes I_{mlN^2} \otimes U_{mlN^2}^{mlN^2 \times mlN^2}\right]\left[I_{mlN^2} \otimes U_{l^2m^2N^4}^{l^2m^2N^4 \times l^2m^2N^4}\right]\right.
\end{aligned}$$

$$\left[(I_{mlN^2} \otimes (\lambda I + L_u^T L_u)^{-1}) \bar{V}_{mlN^2}^{l^2 mN^3 \times mN} \otimes V_{mlN^2}^{lm^2N^3 \times lN} [(\lambda I + L_u^T L_u)^{-1}]^T\right]$$

$$U_{lmN^2}^{lmN^2 \times lmN^2} U_{lmN^2}^{lmN^2 \times lmN^2}\} \tag{6.98}$$

$$F_4 = L_u K \approx N\left(\hat{L}_u \hat{K}, \Sigma_{F_4}\right) \tag{6.99}$$

$$\Sigma_{F_4} = \left[S_{L_u}^T \otimes I_{m^2N^2}\right] \{\left[I_{lmN^2} \otimes U_{m^3lN^4}^{m^3lN^4 \times m^3lN^4}\right]$$

$$\left[(I_{lmN^2} \otimes K^T) V_{lmN^2}^{l^2mN^3 \times mN} \otimes \bar{V}_{lmN^2}^{m^2lN^3 \times lN} K\right] U_{m^2N^2}^{m^2N^2 \times m^2N^2}$$

$$+ \left[I_{lmN^2} \otimes I_{lmN^2} \otimes U_{m^2N^2}^{m^2N^2 \times m^2N^2}\right] \left[I_{lmN^2} \otimes U_{m^3lN^4}^{m^3lN^4 \times m^3lN^4}\right]$$

$$\left[\bar{V}_{mlN^2}^{m^2lN^3 \times lN} K \otimes (I_{lmN^2} \otimes K^T) V_{lmN^2}^{l^2mN^2 \times mN}\right] U_{m^2N^2}^{m^2N^2 \times m^2N^2} U_{m^2N^2}^{m^2N^2 \times m^2N^2}\}$$

$$+ \left[S_K^T \otimes I_{m^2N^2}\right] \{\left[I_{lmN^2} \otimes U_{m^3lN^4}^{m^3lN^4 \times m^3lN^4}\right]$$

$$\left[V_{lmN^2}^{m^2lN^3 \times lN} L_u^T \otimes (I_{lmN^2} \otimes L_u) \bar{V}_{lmN^2}^{m^2lN^3 \times mN}\right] U_{m^2N^2}^{m^2N^2 \times m^2N^2}$$

$$+ \left[I_{mlN^2} \otimes I_{mlN^2} \otimes U_{m^2N^2}^{m^2N^2 \times m^2N^2}\right] \left[I_{lmN^2} \otimes U_{m^3lN^4}^{m^3lN^4 \times m^3lN^4}\right]$$

$$\left[(I_{lmN^2} \otimes L_u) \bar{V}_{lmN^2}^{l^2mN^3 \times mN} \otimes V_{lmN^2}^{m^2lN^3 \times lN} L_u^T\right]$$

$$U_{m^2N^2}^{m^2N^2 \times m^2N^2} U_{m^2N^2}^{m^2N^2 \times m^2N^2}\} \tag{6.100}$$

$$F_5 = I - L_u K \approx N\left([I - \hat{L}_u \hat{K}], \Sigma_{F_4}\right) \tag{6.101}$$

$$LQG_y = (I - L_u K) l_e \approx N\left(L\hat{Q}G_y, \Sigma_{L\hat{Q}G_y}\right) \tag{6.102}$$

$$\Sigma_{L\hat{Q}G_y} = \left[S_{F_5}^T \otimes I_{m^2N}\right] \{\left[I_{m^2N^2} \otimes U_{m^4N^3}^{m^4N^3 \times m^4N^3}\right]$$

$$\left[(I_{m^2N^2} \otimes l_e^T) V_{m^2N^2}^{m^3N^3 \times mN} \otimes \bar{V}_{m^2N^2}^{m^3N^3 \times mN} l_e\right] U_{m^2N}^{m^2N \times m^2N}$$

$$+ \left[I_{m^2N^2} \otimes I_{m^2N^2} \otimes U_{m^2N}^{m^2N \times m^2N}\right] \left[I_{m^2N^2} \otimes U_{m^4N^3}^{m^4N^3 \times m^4N^3}\right]$$

$$\left[\bar{V}_{m^2N^2}^{m^3N^3 \times mN} l_e \otimes (I_{m^2N^2} \otimes l_e^T) V_{m^2N^2}^{m^3N^3 \times mN}\right] U_{m^2N}^{m^2N \times m^2N} U_{m^2N}^{m^2N \times m^2N}\}$$

$$+ \left[S_{l_e}^T \otimes I_{m^2N}\right] \{\left[I_{m^2N} \otimes U_{m^4N^2}^{m^4N^2 \times m^4N^2}\right]$$

$$\left[V_{m^2N}^{m^3N \times mN} F_5^T \otimes (I_{m^2N} \otimes F_5) \bar{V}_{m^2N}^{m^3N^2 \times m}\right] U_{m^2N}^{m^2N \times m^2N}$$

$$+ \left[I_{m^2N} \otimes I_{m^2N} \otimes U_{m^2N}^{m^2N \times m^2N}\right] \left[I_{m^2N} \otimes U_{m^4N^2}^{m^4N^2 \times m^4N^2}\right]$$

$$\left[(I_{m^2N} \otimes F_5) \bar{V}_{m^2N}^{m^3N^2 \times m} \otimes V_{m^2N}^{m^3N \times mN} F_5^T\right] U_{m^2N}^{m^2N \times m^2N} U_{m^2N}^{m^2N \times m^2N}\} \tag{6.103}$$

$$LQG_u = -K l_e \approx N\left(L\hat{Q}G_u, \Sigma_{L\hat{Q}G_u}\right) \tag{6.104}$$

$$\Sigma_{L\hat{Q}G_u} = \left[S_K^T \otimes I_{mlN}\right] \{\left[I_{lmN^2} \otimes U_{l^2m^2N^3}^{l^2m^2N^3 \times l^2m^2N^3}\right]$$

$$\left[(I_{lmN^2} \otimes l_e^T) V_{lmN^2}^{lm^2N^3 \times lN} \otimes \bar{V}_{lmN^2}^{l^2mN^3 \times mN} l_e\right] U_{lmN}^{lmN \times lmN}$$

$$+ \left[I_{lmN^2} \otimes I_{lmN^2} \otimes U_{lmN}^{lmN \times lmN}\right] \left[I_{lmN^2} \otimes U_{l^2m^2N^3}^{l^2m^2N^3 \times l^2m^2N^3}\right]$$

$$\left[\bar{V}_{lmN^2}^{l^2mN^3 \times mN} l_e \otimes (I_{lmN^2} \otimes l_e^T) V_{lmN^2}^{m^2lN^3 \times lN}\right] U_{lmN}^{lmN \times lmN} U_{lmN}^{lmN \times lmN}\}$$

$$+\left[S_{l_e}^T \otimes I_{lmN}\right]\{\left[I_{m^2N} \otimes U_{lm^3N^2}^{lm^3N^2\times lm^3N^2}\right]$$
$$\left[V_{m^2N}^{m^3N\times mN}K^T \otimes (I_{m^2N} \otimes K)\bar{V}_{m^2N}^{m^3N^2\times m}\right]U_{lmN}^{lmN\times lmN}$$
$$+\left[I_{m^2N} \otimes I_{m^2N} \otimes U_{lmN}^{lmN\times lmN}\right]\left[I_{m^2N} \otimes U_{lm^3N^2}^{lm^3N^2\times lm^3N^2}\right]$$
$$\left[(I_{m^2N} \otimes K)\bar{V}_{m^2N}^{m^3N^2\times m} \otimes V_{m^2N}^{m^3N\times mN}K^T\right]U_{m^2N}^{m^2N\times m^2N}U_{m^2N}^{m^2N\times m^2N}\} \quad (6.105)$$

6.8 Simulations

Consider the following system

$$x_{k+1} = \begin{bmatrix} 0.6 & 0.6 & 0 \\ -0.6 & 0.6 & 0 \\ 0 & 0 & 0.7 \end{bmatrix} x_k + \begin{bmatrix} 1.6161 \\ -0.3481 \\ 2.6319 \end{bmatrix} u_{k-1} + \begin{bmatrix} -1.1472 \\ -1.5204 \\ -3.1993 \end{bmatrix} e_k$$
$$y_k = \begin{bmatrix} -0.4373 & -0.5046 & 0.0936 \end{bmatrix} x_k + [-0.7759]u_{k-1} + e_k$$

where x_k, y_k, u_k and e_k represent the system state, output, input and the unmeasured random noise respectively at time k. Open loop process input/output' data is obtained by exciting the system using a designed 'RBS' signal of magnitude 1 for the system input and random white noise of standard deviation 0.1 for e_k in MATLAB-*Simulink*. The open loop data is used to obtain the subspace matrices and their covariance matrices using the methodology described in section (6.6.1). The data Hankel matrices are constructed with rows$(N) = 3$ and columns$(j) = 500$.

The idea of this exercise is to demonstrate the calculations involved in the covariance calculations, hence a smaller number is chosen for the row size, N. The calculations are done for the first three impulse response coefficients of the closed loop noise model if an LQG-controller were implemented on the system. The different steps in obtaining the covariance matrices are shown as follows:

(i) Estimations of $\{\hat{L}_u, \hat{L}_e\}$ and $\{\hat{\Sigma}_{L_u}, \hat{\Sigma}_{L_e}\}$ from open loop process data.

(ii) Estimation of the matrices $\{\hat{L}_u^T, (\hat{L}_u^T\hat{L}_u), [\lambda I + \hat{L}_u^T\hat{L}_u]^{-1}, \hat{K}, (\hat{L}_u\hat{K}), [I - \hat{L}_u\hat{K}],\}$ and their covariance matrices $\{\hat{\Sigma}_{L_u^T}, \Sigma_{(\hat{L}_u^T\hat{L}_u)}, \Sigma_{[\lambda I+\hat{L}_u^T\hat{L}_u]^{-1}}, \Sigma_K, \Sigma_{(\hat{L}_u\hat{K})}, \Sigma_{[I-\hat{L}_u\hat{K}]},\}$

(iii) Estimation of the impulse response coefficients of the noise model LQG_y (closed loop noise model between $y_t \longrightarrow e_t$) and LQG_u (closed loop noise model between $u_t \longrightarrow e_t$), and the covariance matrices of the vector of the impulse response coefficients. For the above simulations the matrices are obtained as follows

$$L\hat{Q}G_y = \begin{bmatrix} 0.9985 \\ 1.3862 \\ 1.0879 \\ \cdots \end{bmatrix}; \quad \Sigma_{L\hat{Q}G_y} = \begin{bmatrix} 0.0184 & 1.1507 & 1.3079 \\ 1.1507 & 1.5497 & 1.0511 \\ 1.3079 & 1.0511 & 0.5300 \end{bmatrix}$$

$$L\hat{Q}G_u = \begin{bmatrix} 0.1439 \\ 0.0854 \\ -0.0048 \end{bmatrix}; \quad \Sigma_{L\hat{Q}G_u} = \begin{bmatrix} 0.4102 & 0.6097 & 0.6120 \\ 0.6097 & 0.6086 & 0.6027 \\ 0.6120 & 0.6027 & 0.4436 \end{bmatrix}$$

Remarks:

Since we have the covariance matrices for the subspace matrices $L\hat{Q}G_y$ and $L\hat{Q}G_u$, it is possible to draw confidence intervals for the LQG-benchmark curve plotted in figure 5.1. However, for the case of higher magnitude of covariance matrices, the confidence limits can be very wide and may not help the original purpose of controller performance assessment using the LQG-benchmark curve. In such case, the data used for identifying the subspace matrices need to be revisited and the excitation signal re-designed so as to obtain smaller magnitude for the covariance matrices.

6.9 Conclusions

Expressions for the statistical characterization of the LQG-benchmark variances have been derived in this chapter. These expressions, although complex, can be used as a guideline for assessing the accuracy of the LQG curve, which completes the controller performance assessment using the LQG benchmark. Simulations are provided for the derivations.

Chapter 7

Performance assessment of multivariate feedback and feedforward controllers without interactor matrix

[1]

7.1 Introduction

Periodic performance assessment of the controllers is important for maintaining normal process operation and to sustaining the performance of controllers achieved when the controllers are commissioned. Controller performance assessment using closed loop data has received much attention over the past decade. Typically, the process response variance is compared with a benchmark variance for assessing performance of the controller. Several benchmarks such as minimum variance control

[1] A version of this chapter has been submitted for publication as a journal paper

R. Kadali, and B. Huang. Multivariate Feedback and Feedforward Control Performance Assessment without Interactor Matrix. *Submitted for publication*, 2002.

(MVC) [5, 19, 20, 28, 34, 36, 38, 42, 46, 50, 51, 83, 111], linear quadratic Gaussian (LQG) control [42, 46, 57, 62, 63, 64], designed controller performance versus achieved controller performance [69, 70, 96], etc. have been proposed for assessing controller performance. Kesavan and Lee [66] proposed a performance diagnosis methodology for multivariate model predictive controllers. Some alternative performance assessment techniques with practical considerations are studied in [22, 28, 33, 85, 100, 116, 117]. Frequency domain analysis has also been used for control loop performance assessment [65]. For a review on the research in this area refer to [38, 46, 98].

Among these approaches MVC-benchmark is one of the popular benchmarks. One of the reasons for the suitability of MVC benchmark to assess performance of control loops in the industry [21, 37, 38, 45, 48, 49, 33, 54, 60, 95, 98, 114, 115, 125] is that it is non intrusive and routine closed loop operating data is sufficient for the calculation of this benchmark.

Minimum variance control is theoretically the best possible control [6]. If a minimum variance controller is implemented on the process, any disturbance entering the process would be attenuated within the process time delay. Controller performance assessment using MVC-benchmark involves comparing the current process output variance with the output benchmark variance if a minimum variance controller were implemented on the process. Although the intention of many industrial controllers is not minimum variance control, MVC-benchmark is used as a first step in the controller performance assessment [46, 60]. Designing a minimum variance controller for univariate systems involves inverse of the delay free part of the process transfer function [6, 38, 42, 46]. Hence the calculation of the MVC-benchmark variance for univariate systems from routine closed loop data requires *a priori* knowledge of the process time delay [19, 20, 36, 37, 38, 42, 46, 111]. However, non-minimum phase systems may require complete process transfer function knowledge for the calculation of the MVC-benchmark variance [46, 116]. For multivariate systems, minimum variance controller contains the inverse of the delay-free part of the process transfer function matrix, which is calculated as the transfer function matrix premultiplied by

the interactor matrix [36, 38, 42, 46]. Interactor matrix in the multivariate case is equivalent to the concept of time-delay in the univariate case (refer to [44, 47, 109]). *A priori* knowledge of the first few process Markov parameters is required for the calculation of the interactor matrix [36, 42, 46]. Hence calculation of MVC-benchmark variance is not straightforward in the multivariate case. Furthermore, the concept of the interactor is not well known in practice. Hence, estimation of the MVC-benchmark without the interactor matrix has been an active area of research.

Ko and Edgar [68] proposed a method for the estimation of the multivariate MVC-benchmark using closed loop data, which does not require the intermediate interactor matrix calculation. However, along with the knowledge of the first few Markov parameters of the process model, this method also requires knowledge of the first few Markov parameters of the noise model and the controller model. Ko and Edgar provided two different approaches for the estimation of the noise model and computation of the Markov parameters from closed loop data.

Recent results in the subspace approach to closed loop identification [58] inspires an alternative approach for the estimation of multivariate MVC-benchmark. In this chapter we will show the estimation of the multivariate MVC-benchmark with *neither* the interactor matrix calculation *nor* the Markov parameters. The only *a priori* knowledge required is the deterministic subspace matrix directly calculated from data. The important difference between the "calculation of the subspace matrix" and subspace identification is that the former does not extract an explicit "model" and is also known as model-free approach in the literature. This will further simplify the procedure for the calculation of the multivariate performance index. No concepts such as interactor matrix, Markov parameter, multivariate transfer function matrix, state space model etc. are needed to apply this technique and this will make the multivariate controller performance assessment technique more applicable in practice.

Subspace identification methods allow the direct identification of a state space model for the system from the data. Certain subspace matrices are identified as

an intermediate step in the subspace identification methods, which correspond to the system states, the deterministic inputs and the stochastic inputs. Normally, subspace identification proceeds in two steps. The first step is a data projection step to obtain the subspace matrices; the second step is to extract the state space matrices from the subspace matrices. However, as will be shown in this chapter, the minimum variance controller can be designed directly using the intermediate subspace matrices, without a parametric model such as the state space model. In fact, the subspace matrices corresponding to the deterministic and stochastic subspace matrices implicitly contain the Markov parameters of the process and noise models respectively. The closed loop subspace identification method proposed by Kadali and Huang [58] allows the identification of the deterministic and stochastic subspace matrices from closed loop experiments directly. As will be shown in the later sections of this chapter the MVC-benchmark variance calculation requires the knowledge of only the deterministic subspace matrix, and therefore provides a new approach for obtaining the MVC-benchmark variance from routine closed loop data and eliminates the intermediate step of estimating the unitary interactor matrix or extracting the Markov parameters.

Most recently, McNabb and Qin [84] proposed an interesting subspace approach to multivariate performance monitoring by projecting interactor (or equivalent) filtered output data onto past data, but their method, once again, requires a delay matrix that is equivalent to the interactor matrix.

Essentially all methods proposed for multivariable control performance assessment need either a process model or an identification experiment. In this sense, there is no fundamental difference in terms of *a priori* knowledge required among all existing approaches including the one proposed in this chapter. However, what are advantages without using the interactor matrix? The interactor matrix is typically calculated from Markov parameters and/or the transfer function matrix. The first advantage is that the model structure error is avoided without forcing the process fit into a limited order parametric model. The second advantage is the conceptual simplicity. As the performance assessment technique is mainly used by practitioners,

simplicity is a key to its acceptance.

The main contributions of this chapter are: (i) designing a minimum variance controller using subspace matrices without an explicit parametric model, (ii) obtaining the multivariate MVC-benchmark variance directly from data without the interactor matrix or Markov parameters, (iii) rigorous proof of the equivalence of the proposed approach and the conventional approach while almost all existing results show the equivalence through simulations only, and (iv) extension of the results to performance assessment of feedforward controllers.

The remaining of this chapter is organized as follows. Section 7.2 describes a method for designing a minimum variance controller from the subspace matrices. Section 7.3 is the main section which provides the method for the estimation of multivariate MVC-benchmark without the interactor matrix. Comparison of the proposed approach with the conventional transfer function approach for obtaining MVC-benchmark variance is described in section 7.4. The proposed subspace based data driven approach is extended to the feedforward control in section 7.5. Algorithm for obtaining the MVC-benchmark variance from routine process operating data is presented in section 7.6. The main results are illustrated though a numerical example in section 7.7 followed by conclusions in section 7.8.

7.2 Design of minimum variance control using subspace matrices

The minimum variance controller (MVC) is designed to minimize the following quadratic cost function J over the horizon N, as $N \longrightarrow \infty$:

$$J = E\{ \sum_{k=1}^{N} [(r_{t+k} - y_{t+k})^T (r_{t+k} - y_{t+k})]\} \quad (7.1)$$

$$= \sum_{k=1}^{N} [(r_{t+k} - \hat{y}_{t+k})^T (r_{t+k} - \hat{y}_{t+k})] \quad (7.2)$$

where E is the expectancy operator, r_t is the reference for output trajectory. $\hat{y}_{t+k}$ is the k-step ahead predicted output given the past inputs and outputs upto time t.

Using equation (2.11), the optimal predictor equation can be written as:

$$\hat{y}_f = L_w w_p + L_u u_f \tag{7.3}$$

where

$$\hat{y}_f = \begin{bmatrix} \hat{y}_{t+1} \\ \dots \\ \hat{y}_{t+N} \end{bmatrix}; u_f = \begin{bmatrix} u_{t+1} \\ \dots \\ u_{t+N} \end{bmatrix}; w_p = \begin{bmatrix} y_p \\ u_p \end{bmatrix}; y_p = \begin{bmatrix} y_{t-N+1} \\ \dots \\ y_t \end{bmatrix}; u_p = \begin{bmatrix} u_{t-N+1} \\ \dots \\ u_t \end{bmatrix}$$

The notation in the cost function can be simplified for regulatory control, by letting $r_{t+k} = 0$, as:

$$\begin{aligned} J &= \min_{u_f^2} \; [\hat{y}_f^T \; \hat{y}_f] \\ &= (L_w w_p + L_u u_f)^T (L_w w_p + L_u u_f) \end{aligned} \tag{7.4}$$

To obtain the minimum variance control law, we differentiate J with respect to u_f and set it to zero.

$$\frac{\partial J}{\partial u_f} = 2L_u^T L_w w_p + 2L_u^T L_u u_f = 0 \tag{7.5}$$

We obtain the control law as:

$$u_f = -L_u^{\dagger}(L_w w_p) \tag{7.6}$$

where, $\dagger$ represents pseudo-inverse. The above control law is the minimum variance control law as the number of block-rows in the subspace matrices L_w and L_u tend to infinity.

7.3 Estimation of the Multivariate MVC-benchmark

From the very first block-element of Y_f in equation (2.11) we can write

$$y_{t+1} = L_{y_p} \begin{bmatrix} y_{t-N+1} \\ \dots \\ y_t \end{bmatrix} + L_{u_p} \begin{bmatrix} u_{t-N+1} \\ \dots \\ u_t \end{bmatrix} + L_0 \; e_{t+1} \tag{7.7}$$

where

$$L_{y_p} = L_w(1:m, 1:mN) \tag{7.8}$$

$$L_{u_p} = L_w(1:m, mN+1:(l+m)N) \tag{7.9}$$

where the notation $A(i:j, p:q)$ represents the rows i to j and columns p to q of the matrix A. Equation (7.7) can be transformed (See appendix D) for an equivalent expression of y_{t+1} in terms of the past inputs and the past noise as

$$y_{t+1} = \begin{bmatrix} G_1 & \dots & G_N \end{bmatrix} \begin{bmatrix} u_t \\ \dots \\ u_{t-N+1} \end{bmatrix} + \begin{bmatrix} L_1 & \dots & L_N \end{bmatrix} \begin{bmatrix} e_t \\ \dots \\ e_{t-N+1} \end{bmatrix} + L_0 \; e_{t+1} \tag{7.10}$$

where G_i and L_i are the i-th impulse response coefficients (Markov parameters for multivariate systems) of the process and noise models respectively. In other words, we can express the past (state) contribution term, $L_w w_p$, as

$$L_w w_p = \begin{bmatrix} G_1 & \dots & G_{N-1} & G_N \\ G_2 & \dots & G_N & 0 \\ \dots & \dots & \dots & \dots \\ G_N & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} u_t \\ \dots \\ u_{t-N+1} \end{bmatrix} + \begin{bmatrix} L_1 & \dots & L_{N-1} & L_N \\ L_2 & \dots & L_N & 0 \\ \dots & \dots & \dots & \dots \\ L_N & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} e_t \\ \dots \\ e_{t-N+1} \end{bmatrix} \tag{7.11}$$

However, the controller output, u_{t+1} is calculated using all the data available at time '$t+1$', i.e., $\{u_t, y_{t+1}, u_{t-1}, y_t, \dots\}$. Hence the original subspace predictor expression in equation (2.11) and the subspace based minimum variance control law in equation (7.6) have to be modified to obtain the closed loop expressions for u_f and y_f. First,

define

$$L_G = \begin{bmatrix} G_1 & G_2 & \dots & G_{N-1} & G_N \\ G_2 & G_3 & \dots & G_N & 0 \\ \dots & \dots & \dots & \dots & \\ G_N & 0 & 0 & \dots & 0 \end{bmatrix}; \quad \tilde{u}_p = \begin{bmatrix} u_t \\ u_{t-1} \\ \dots \\ u_{t-N+1} \end{bmatrix}$$

$$L_H = \begin{bmatrix} L_0 & L_1 & \dots & L_{N-1} & L_N \\ L_1 & L_2 & \dots & L_N & 0 \\ \dots & \dots & \dots & \dots & \dots \\ L_{N-1} & 0 & 0 & \dots & 0 \end{bmatrix}; \quad \tilde{e}_p = \begin{bmatrix} e_{t+1} \\ e_t \\ \dots \\ e_{t-N+1} \end{bmatrix}$$

$$\tilde{L}_e = \begin{bmatrix} 0 & 0 & \dots & 0 \\ L_0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ L_{N-2} & L_{N-3} & \dots & 0 \end{bmatrix}; \quad \tilde{e}_f = \begin{bmatrix} e_{t+2} \\ e_{t+3} \\ \dots \\ e_{t+N+1} \end{bmatrix}$$

Since L_G and L_H contain the process and noise model Markov parameters, they can be formed from the subspace matrices L_u and L_e respectively. Therefore the equation based on the first column of Y_f in equation (2.11) can be alternatively written as

$$y_f = L_G \tilde{u}_p + L_H \tilde{e}_p + L_u u_f + \tilde{L}_e \tilde{e}_f \qquad (7.12)$$

Substituting equation (7.11) in equation (7.6), we can write

$$u_f = -L_u^{\dagger} \{L_w w_p\} = -L_u^{\dagger} \{L_G \tilde{u}_p + L_H \tilde{e}_p\} \qquad (7.13)$$

The closed loop expression for y_f can be written as

$$y_f = (I - L_u L_u^{\dagger}) (L_G \tilde{u}_p + L_H \tilde{e}_p) + \tilde{L}_e \tilde{e}_f \qquad (7.14)$$

Now that we have derived closed-loop expressions for both u and y, the next step is to calculate their variance expressions which are actually the H_2 norm of the closed-loop expressions weighted by the variance of e. A simple method to derive the variance expression is given below.

Let a disturbance enter the process at time $= t+1$, i.e.,

$$\begin{aligned} u_t &= u_{t-1} = ... = u_{t-N+1} = 0 \\ e_t &= e_{t-1} = ... = e_{t-N+1} = 0 \\ e_{t+2} &= e_{t+3} = ... = e_{t+N} = 0 \end{aligned}$$

Then the cumulative effect of the noise e_{t+1} on the process output variance can be obtained from equation (7.14), which simplifies to

$$y_f = (I - L_u L_u^{\dagger}) L_h \; e_{t+1} = \begin{bmatrix} \psi_0 \\ ... \\ \psi_{d-1} \\ 0 \\ ... \end{bmatrix} e_{t+1} = \Psi e_{t+1} \tag{7.15}$$

where $L_h = \begin{bmatrix} L_0 \\ ... \\ L_{N-1} \end{bmatrix}$, the vector of noise model Markov parameters, and ψ_i represents the Markov parameter of i-th lag of the closed loop noise model if a minimum variance controller were implemented on the system described in equations (2.1)-(2.2). 'd' is the order of the interactor matrix for the system (2.1)-(2.2) and is unique for a given system [89, 109]. The variance of the closed-loop system can be calculated from the Markov parameters/impulse response of the closed-loop system and the minimum variance control variance expression for the process output is given by

$$var[y_t]_{MVC} = \sum_{i=0}^{d-1} \psi_i var[e_t] \psi_i^T \tag{7.16}$$

Note that estimation of the interactor matrix is *not* required for obtaining the MVC-benchmark variance. However the above result requires the knowledge of $L_h = \begin{bmatrix} L_0 \\ ... \\ L_{N-1} \end{bmatrix}$, and hence it appears that estimation of the noise model in the Markov parameters model is necessary.

However, we will show that the estimation of $\begin{bmatrix} L_0 \\ ... \\ L_{N-1} \end{bmatrix}$ is *not* required. The closed loop noise model Markov parameters $L_h^{CL} = \begin{bmatrix} L_0^{CL} \\ ... \\ L_{N-1}^{CL} \end{bmatrix}$ (the vector of closed-loop noise model which can be estimated from the routine operating data) can be used in the place of $\begin{bmatrix} L_0 \\ ... \\ L_{N-1} \end{bmatrix}$ and we can still be able to obtain the MVC-benchmark variance, where $L_h^{CL} = (I + L_u L_c)^{-1} L_h$ (refer to chapter 3).

where $L_h = \begin{bmatrix} L_0 \\ ... \\ L_{N-1} \end{bmatrix}$, the vector of noise model Markov parameters, and ψ_i represents the Markov parameter of i-th lag of the closed loop noise model if a minimum variance controller were implemented on the system described in equations (2.1)-(2.2). 'd' is the order of the interactor matrix for the system (2.1)-(2.2) and is unique for a given system [89, 109]. The variance of the closed-loop system can be calculated from the Markov parameters/impulse response of the closed-loop system and the minimum variance control variance expression for the process output is given by

$$var[y_t]_{MVC} = \sum_{i=0}^{d-1} \psi_i var[e_t] \psi_i^T \tag{7.17}$$

Lemma 1: *Ψ can be obtained using the vector of Markov parameters of the closed loop noise model, L_h^{CL}, in place of the L_h in equation (7.15) .*

Proof: The above statement is equivalent to saying that $(I - L_u L_u^{\dagger}) L_h$ and $(I - L_u L_u^{\dagger}) L_h^{CL}$ yield the same result. Now,

$$(I - L_u L_u^{\dagger}) L_h^{CL} = (I - L_u L_u^{\dagger}) \ (I + L_u L_c)^{-1} \ L_h \tag{7.18}$$

Therefore on observation, we need to show that

$$(I - L_u L_u^{\dagger}) \;=\; (I - L_u L_u^{\dagger})\;(I + L_u L_c)^{-1} \tag{7.19}$$

to prove the lemma, which is equivalent to showing

$$(I - L_u L_u^{\dagger})\;(I + L_u L_c) \;=\; (I - L_u L_u^{\dagger}) \tag{7.20}$$

Expanding the left hand side term in the above equation

$$\begin{aligned}(I - L_u L_u^{\dagger})\;(I + L_u L_c) \;&=\; I + L_u L_c - L_u L_u^{\dagger} - L_u L_u^{\dagger} L_u L_c && (7.21)\\ &=\; I - L_u L_u^{\dagger} + \{L_u L_c - L_u L_u^{\dagger} L_u L_c\} && (7.22)\\ &=\; I - L_u L_u^{\dagger} && (7.23)\end{aligned}$$

The last equation follows since $L_u L_u^{\dagger} L_u = L_u$.

Lemma 1 is essentially the subspace version of the invariance property of the first few Markov parameters of the interactor-filtered noise model under the transfer function framework originally derived in Huang and Shah (1999)[46]. This invariance property has also been proved in Ko and Edgar (2001)[68].

Hence the Markov parameters of the closed loop noise model can be used in place of Markov parameters of the open loop noise model and we can still get the same benchmark variance. Therefore we need *only* the subspace matrix L_u (which contains Markov parameters of the process and is estimated from data) for the calculation of the minimum variance control benchmark. The Markov parameters of the closed loop noise model (or noise subspace matrix) can be easily estimated from the routine operating data as shown below:

For routine process operating data i.e., closed-loop data with no set point excitation, the past data Hankel matrix is taken as $M_p = \begin{bmatrix} Y_p \\ U_p \end{bmatrix}$. The subspace expression from equation (3.14) in chapter 3 becomes

$$\begin{aligned}Y_f \;&=\; L_y^{CL} M_p + L_{ye}^{CL} E_f && (7.24)\\ & && (7.25)\end{aligned}$$

and

$$\hat{Y}_f = L_y^{CL} M_p \tag{7.26}$$

Using linear regression L_y^{CL} and $\hat{Y}_f$ can be estimated. The first block-row of $Y_f - \hat{Y}_f = L_{ye}^{CL} E_f$ represents the one-step ahead prediction errors of the process output. Therefore the stochastic disturbance sequence entering the process can be estimated as

$$e_f = \begin{bmatrix} e_{N+1} & e_{N+2} & \dots & e_{N+j} \end{bmatrix} = Y_f(1:m,:) - \hat{Y}_f(1:m,:) \tag{7.27}$$

Using the estimated noise e_f the data block-Hankel matrix E_f can be formed in the same manner as shown in equations (2.4)-(2.5). The noise model subspace matrix, L_{ye}^{CL}, can therefore be estimated as

$$L_{ye}^{CL} = \left[Y_f - \hat{Y}_f\right] / E_f \tag{7.28}$$

The matrix L_h^{CL}, required for MVC-benchmark calculation, can be obtained from L_{ye}^{CL} as

$$L_h^{CL} = L_{ye}^{CL}(:, 1:m) \tag{7.29}$$

7.4 Equivalence of subspace approach to the conventional transfer function approach in obtaining the MVC-benchmark variance

For a multivariate system with l-inputs and m-outputs, the subspace matrices $L_u(mN \times lN)$ and $L_e(mN \times mN)$ contain the Markov parameters of the deterministic and stochastic inputs respectively. Markov parametric matrices are equivalent to the impulse response coefficients for a univariate system. The matrices L_u and L_e would be

$$L_u = \begin{bmatrix} G_0 & 0 & \dots & 0 \\ G_1 & G_0 & 0 & \dots \\ \dots & \dots & \dots & \dots \end{bmatrix}; \quad L_e = \begin{bmatrix} L_0 & 0 & \dots & 0 \\ L_1 & L_0 & \dots & 0 \\ \dots & \dots & \dots & \dots \end{bmatrix}$$

It is implicit that there is at least one time-delay i.e., $G_0 = 0$. Due to the time delays, L_u is rank deficient and hence singular. Now, L_u consists of two components, a non-invertible component and an invertible component. Pseudo-inverse of L_u inverts only the invertible part of L_u. Hence the term $(I - L_u L_u^\dagger)$ represents the contribution of 'controller invariance' to the process output variance. The analogies to the transfer function domain approach are obvious here.

In the transfer function domain, a unitary interactor matrix is first estimated using the process Markov parameters. The multiplication of the interactor matrix and the transfer function matrix represents the invertible part of the transfer function matrix. Let d be the order of the unitary interactor matrix if it were calculated using the process Markov parameters [89, 109]. The vectors of Markov parameters in L_u are *essentially disjoint* matrices (see appendix F for the definition). In subspace approach d represents the number of Markov parameters which make the matrix $\begin{bmatrix} G_0 \\ G_1 \\ \cdots \\ G_i \end{bmatrix}$ rank deficient till $i < d$ and a full rank matrix at $i \geq d$. Hence the matrix $(I - L_u L_u^\dagger)$ will have zeros below the rows $> (dm)$. Note that the order 'd' does not have to be estimated in the subspace approach. We mention it only for illustrating the equivalence of both approaches.

Let us look at the structure and elements of the different intermediate matrices in the subspace approach of MVC-benchmark estimation:

$$L_u^\dagger = \begin{bmatrix} 0_{mN \times kl} & P_{mN \times (N-k)l} \end{bmatrix} \tag{7.30}$$

$$L_u L_u^\dagger = \begin{bmatrix} 0_{km \times km} & 0_{km \times (d-k)m} & 0 & \cdots \\ 0_{(d-k)m \times km} & M_{(d-k)m \times (d-k)m} & 0 & \cdots \\ 0_{(N-d)m \times km} & 0_{(N-d)m \times (d-k)m} & I_{(N-d)m} & \end{bmatrix} \tag{7.31}$$

$$I - L_u L_u^\dagger = \begin{bmatrix} I_{km} & 0_{km \times (d-k)m} & 0 & \cdots \\ 0_{(d-k)m \times km} & \left[I_{(d-k)m} - M_{(d-k)m \times (d-k)m} \right] & 0 & \cdots \\ 0_{(N-d)m \times km} & 0_{(N-d)m \times (d-k)m} & 0 & \cdots \end{bmatrix} \tag{7.32}$$

where $k \leq d$ is the number of the first few Markov parameters that are zero matrices. P and M are internal matrices in the calculations. Therefore we obtain

$$\Psi = \begin{bmatrix} I_{km} & 0_{km\times(d-k)m} & 0 & \dots \\ 0_{(d-k)m\times km} & \left[I_{(d-k)m} - M_{(d-k)m\times(d-k)m}\right] & 0 & \dots \\ 0 & 0 & 0 & \dots \\ \dots & \dots & \dots & \dots \end{bmatrix} \begin{bmatrix} L_0 \\ L_1 \\ \dots \\ L_N \end{bmatrix} \quad (7.33)$$

Thus only the first d terms remain for the calculation of the MVC variance of the process outputs. From the above equation we can see that the controller invariant part of the noise model for calculating the MVC-benchmark variance of the process output is directly obtained from the subspace matrices. In other words, the subspace based approach 'directly' gives the interactor filtered Markov parameters of the noise model. Hence the calculation of the interactor matrix is eliminated.

In the transfer function approach, the transfer function matrix and the interactor matrix for a system can be written (with some abuse of notation used in [46]) as follows:

$$G(z^{-1}) = G_0 + G_1 z^{-1} + G_2 z^{-2} + \dots \quad (7.34)$$

$$D(z) = D_1 z + D_2 z^2 + ... + D_{d-1} z^{d-1} + D_d z^d \quad (7.35)$$

The condition for the interactor matrix from theorem 3.2.1 in [46] is

$$\lim_{z^{-1}\longrightarrow 0} DG = \lim_{z^{-1}\longrightarrow 0} \left[D_d z^d + \dots + D_1 z\right]\left[G_1 z^{-1} + G_2 z^{-2} + \dots\right] = K_{mat} \quad (7.36)$$

where T is a full rank matrix. Therefore we have

$$\begin{aligned} D_d G_1 &= 0 \\ D_{d-1} G_1 + D_d G_2 &= 0 \\ \dots &= \dots \\ D_1 G_1 + D_2 G_2 + \dots + D_d G_d &= K_{mat} \end{aligned}$$

The above set of equations can be alternatively expressed as two conditions

Condition-1

$$\begin{bmatrix} D_1 & D_2 & \dots & D_d \end{bmatrix} \begin{bmatrix} G_0 & 0 & \dots & 0 \\ G_1 & G_0 & \dots & 0 \\ \dots & \dots & \dots & 0 \\ G_{d-1} & G_{d-2} & \dots & G_0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & \dots & 0 \end{bmatrix} \tag{7.37}$$

Condition-2

$$\begin{bmatrix} D_1 & D_2 & \dots & D_d \end{bmatrix} \begin{bmatrix} G_1 \\ G_2 \\ \dots \\ G_d \end{bmatrix} = K_{mat} \tag{7.38}$$

with $K_{mat} = min\{m, l\}$.

We need to show that the coefficients obtained in the subspace approach are same as those obtained in the transfer function domain approach, i.e. the above two conditions are satisfied by using the matrix $\left(I - L_u \; L_u^{\dagger}\right)$. Therefore we have to prove the following theorem for the subspace approach:

Theorem 7.1 $\left(I - L_u \; L_u^{\dagger}\right)$ *contains interactor matrix for the process. An interactor matrix can be constructed directly from this expression. The subspace approach for the calculation of the minimum variance control benchmark is equivalent to that of the conventional transfer function approach.*

Proof: Consider the singular value decomposition of the matrix L_u which produces a diagonal matrix Σ, of the same dimension as L_u and with nonnegative diagonal elements in decreasing order, and unitary matrices (satisfies the property $(.)^T = (.)^{-1}$) Φ and Θ so that

$$L_u \;=\; \Phi \; \Sigma \; \Theta^T = \begin{bmatrix} \Phi_1 & \Phi_2 \end{bmatrix} \begin{bmatrix} \Sigma_r & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Theta_1^T \\ \Theta_2^T \end{bmatrix} = \Phi_1 \; \Sigma_r \; \Theta_1^T \tag{7.39}$$

where r is the number of singular values of L_u that are not zero. The matrices Φ and Θ are orthogonal matrices (the columns are mutually orthogonal vectors of unit length, i.e. $\Phi(:,i)^T\Phi(:,j) = 0$ for $i \neq j$ and 1 for $i = j$) and are non-unique:

(a) Φ_1 spans the range space of L_u, and

(b) Φ_2 spans the null space of L_u.

We can write

$$L_u^{\dagger} \;=\; \Theta_1 \; \Sigma_r^{-1} \; \Phi_1^T \;= \Phi_1 \; \Sigma_r^{-1} \; \Theta_1^T \tag{7.40}$$

Therefore we can obtain the simplified expressions (see appendix E for details)

$$L_u \; L_u^{\dagger} \;=\; \Phi_1 \; \Phi_1^T \tag{7.41}$$

$$I - L_u \; L_u^{\dagger} \;=\; \Phi_2 \; \Phi_2^T \tag{7.42}$$

Φ_2 spans the null space of the matrix L_u. Therefore, $\Phi_2 \Phi_2^T L_u = 0$, i.e.,

$$\left(I - L_u \; L_u^{\dagger}\right) \begin{bmatrix} G_0 & 0 & \ldots & 0 \\ G_1 & G_0 & 0 & \ldots \\ \ldots & \ldots & \ldots & \ldots \end{bmatrix} \;=\; \left(I - L_u \; L_u^{\dagger}\right) L_u = 0 \tag{7.43}$$

From the above equation condition-1 expressed in equation (7.37) is satisfied.

Next, consider the transformed Markov parameter matrix $\begin{bmatrix} \hat{G}_1 \\ \ldots \\ \hat{G}_d \end{bmatrix} = (I - L_u L_u^{\dagger}) \begin{bmatrix} G_1 \\ \ldots \\ G_d \end{bmatrix}$. Note that the matrices L_u and $\begin{bmatrix} G_1 \\ \ldots \\ G_d \end{bmatrix}$ are *essentially disjoint* (see appendix F). Following the corollary 17.2.10 (see appendix G) in ref.[39]

$$rank \begin{bmatrix} \hat{G}_1 \\ \ldots \\ \hat{G}_d \end{bmatrix} \;=\; rank \left\{ (I - L_u L_u^{\dagger}) \begin{bmatrix} G_1 \\ \ldots \\ G_d \end{bmatrix} \right\} = rank \begin{bmatrix} G_1 \\ \ldots \\ G_d \end{bmatrix} \tag{7.44}$$

Now let

$$K_{mat} \;=\; \hat{G}_1 \;+\; \ldots \;+\; \hat{G}_d = \begin{bmatrix} I_m & \ldots & I_m \end{bmatrix} \begin{bmatrix} \hat{G}_1 \\ \ldots \\ \hat{G}_d \end{bmatrix} \tag{7.45}$$

Using the corollary 17.5.2 (see appendix G) from [39] we can write

$$rank\{K_{mat}\} = rank\left\{\left[\begin{array}{ccc} I_m & \ldots & I_m \end{array}\right]\right\} + rank\left\{\left\{\left[\begin{array}{c} \hat{G}_1 \\ \ldots \\ \hat{G}_d \end{array}\right]\right\} - (dm)\right.$$

$$+rank\left\{\left(\left(I_{dm} - \left[\begin{array}{c} \hat{G}_1 \\ \ldots \\ \hat{G}_d \end{array}\right]\left[\begin{array}{c} \hat{G}_1 \\ \ldots \\ \hat{G}_d \end{array}\right]^{\dagger}\right)\left(I_{dm} - \left[\begin{array}{ccc} I_m & \ldots & I_m \end{array}\right]^{\dagger}\left[\begin{array}{ccc} I_m & \ldots & I_m \end{array}\right]\right)\right\} \tag{7.46}$$

The matrix $\left[\begin{array}{ccc} I_m & \ldots & I_m \end{array}\right]$ is $(m \times dm)$ dimensional and $rank(\left[\begin{array}{ccc} I_m & \ldots & I_m \end{array}\right]) = m$.

$rank\left\{\left(\left(I_{dm} - \left[\begin{array}{c} \hat{G}_1 \\ \ldots \\ \hat{G}_d \end{array}\right]\left[\begin{array}{c} \hat{G}_1 \\ \ldots \\ \hat{G}_d \end{array}\right]^{\dagger}\right)\left(I_{dm} - \left[\begin{array}{ccc} I_m & \ldots & I_m \end{array}\right]^{\dagger}\left[\begin{array}{ccc} I_m & \ldots & I_m \end{array}\right]\right)\right\}$ is determined in the following manner:

Consider $\mathcal{A} = \left[\begin{array}{ccc} I_m & \ldots & I_m \end{array}\right]_{m \times dm}$ and $\mathcal{B} = \left[\begin{array}{c} \hat{G}_1 \\ \ldots \\ \hat{G}_d \end{array}\right]_{dm \times l}$. Therefore,

$$(I - \mathcal{B}\mathcal{B}^{\dagger})(I - \mathcal{A}^{\dagger}\mathcal{A}) = (I - \mathcal{A}^{\dagger}\mathcal{A}) - \mathcal{B}^{\dagger}\mathcal{B}(I - \mathcal{A}^{\dagger}\mathcal{A}) \tag{7.47}$$

For using the item (3) in appendix G, we take,

$$\mathcal{R} = (I - \mathcal{A}^{\dagger}\mathcal{A}); \quad \mathcal{S} = -\mathcal{B}; \quad \mathcal{T} = \mathcal{B}^{\dagger}; \quad \mathcal{U} = (I - \mathcal{A}^{\dagger}\mathcal{A}) \tag{7.48}$$

we write

$$\mathcal{E}_{\mathcal{R}} = I - \mathcal{R}\mathcal{R}^{\dagger} = \mathcal{A}^{\dagger}\mathcal{A} \tag{7.49}$$

$$\mathcal{F}_{\mathcal{R}} = I - \mathcal{R}^{\dagger}\mathcal{R} = \mathcal{A}^{\dagger}\mathcal{A} \tag{7.50}$$

$$\mathcal{X} = \mathcal{E}_{\mathcal{R}}\mathcal{S}\mathcal{T} = \mathcal{A}^{\dagger}(\mathcal{A}\mathcal{B}\mathcal{B}^{\dagger}) \tag{7.51}$$

$$\mathcal{Y} = \mathcal{T}\mathcal{U}\mathcal{F}_{\mathcal{R}} = \mathcal{B}^{\dagger}(I - \mathcal{A}^{\dagger}\mathcal{A})\mathcal{A}^{\dagger}\mathcal{A} = 0 \tag{7.52}$$

$$\mathcal{Q} = \mathcal{T} + \mathcal{U}\mathcal{R}^{\dagger}\mathcal{S}\mathcal{T} = (\mathcal{A}\mathcal{B})^{\dagger}(\mathcal{A}\mathcal{B})\mathcal{B}^{\dagger} \tag{7.53}$$

$$\mathcal{M} = \mathcal{X}(I - \mathcal{Q}^{\dagger}\mathcal{Q}) = \mathcal{A}^{\dagger}(\mathcal{A}\mathcal{B}\mathcal{B}^{\dagger})\left[I - \mathcal{B}(\mathcal{A}\mathcal{B})^{\dagger}(\mathcal{A}\mathcal{B})(\mathcal{A}\mathcal{B})^{\dagger}(\mathcal{A}\mathcal{B}\mathcal{B}^{\dagger})\right]$$

$$= \mathcal{A}^\dagger(\mathcal{A}\mathcal{B}\mathcal{B}^\dagger)\left[I - \mathcal{B}(\mathcal{A}\mathcal{B})^\dagger(\mathcal{A}\mathcal{B}\mathcal{B}^\dagger)\right] = \mathcal{A}^\dagger(\mathcal{A}\mathcal{B}\mathcal{B}^\dagger)\left[I - (\mathcal{A}\mathcal{B}\mathcal{B}^\dagger)^\dagger(\mathcal{A}\mathcal{B}\mathcal{B}^\dagger)\right]$$

$$= 0 \tag{7.54}$$

$$\mathcal{N} = (I - \mathcal{Q}\mathcal{Q}^\dagger)\mathcal{Y} = 0 \tag{7.55}$$

In the above equations we have taken $(I - \mathcal{A}^\dagger\mathcal{A})(I - \mathcal{A}^\dagger\mathcal{A})^\dagger = (I - \mathcal{A}^\dagger\mathcal{A})$, since $(I - \mathcal{A}^\dagger\mathcal{A})$ is an idempotent matrix and $(*)^\dagger(*)(*)^\dagger = (*)^\dagger$. We can write

$$rank[\mathcal{R}] = rank[(I - \mathcal{A}^\dagger\mathcal{A})] \tag{7.56}$$

$$rank[\mathcal{Q}] = rank[(\mathcal{A}\mathcal{B})^\dagger(\mathcal{A}\mathcal{B})\mathcal{B}^\dagger] \tag{7.57}$$

$$rank[\mathcal{M}] = 0 \tag{7.58}$$

$$rank[\mathcal{N}] = 0 \tag{7.59}$$

$$rank[\mathcal{T}] = rank[\mathcal{B}^\dagger] \tag{7.60}$$

and

$$rank\left[(I - \mathcal{M}\mathcal{M}^\dagger)\mathcal{X}\mathcal{Q}^\dagger\mathcal{Y}(I - \mathcal{N}^\dagger\mathcal{N})\right] = 0 \tag{7.61}$$

Using equations (7.46) and (G.2), we write,

$$rank[K_{mat}] = rank[\mathcal{A}] + rank[\mathcal{B}] - dm + rank[(I - \mathcal{A}^\dagger\mathcal{A})]$$
$$+rank[(\mathcal{A}\mathcal{B})^\dagger(\mathcal{A}\mathcal{B})\mathcal{B}^\dagger] - rank[\mathcal{B}^\dagger] \tag{7.62}$$

$$= m + rank[\mathcal{B}] - dm + (d-1)m + rank[(\mathcal{A}\mathcal{B})^\dagger(\mathcal{A}\mathcal{B})\mathcal{B}^\dagger] - rank[\mathcal{B}^\dagger] \tag{7.63}$$

$$= rank[(\mathcal{A}\mathcal{B})^\dagger(\mathcal{A}\mathcal{B})\mathcal{B}^\dagger] \tag{7.64}$$

In the above equation we used $rank[\mathcal{B}] = rank[\mathcal{B}^\dagger]$ and $rank[(I - \mathcal{A}^\dagger\mathcal{A})] = (d-1)m$. Consider the two cases,

(i) $m \geq l$

In this case $(\mathcal{A}\mathcal{B})^\dagger(\mathcal{A}\mathcal{B}) = I_l$. Therefore $rank[(\mathcal{A}\mathcal{B})^\dagger(\mathcal{A}\mathcal{B})\mathcal{B}^\dagger] = rank[\mathcal{B}^\dagger] = l$ and

$$rank[K_{mat}] = l \tag{7.65}$$

(ii) $m < l$

In this case $rank[(\mathcal{AB})^{\dagger}(\mathcal{AB})] = m$.

Since $\mathcal{B}$ is a full rank matrix $rank[(\mathcal{AB})^{\dagger}(\mathcal{AB})\mathcal{B}^{\dagger}] = m$. Therefore

$$rank[K_{mat}] \;=\; m \tag{7.66}$$

Hence K_{mat} is a full rank matrix and condition-2 expressed in equation (7.38) is satisfied.

Hence the theorem is proved.

Therefore the matrix $\left(I - L_u \; L_u^{\dagger}\right)$ performs the same function as an interactor matrix in the transfer function domain. But the calculation of interactor matrix is not required in deriving the MVC-benchmark variance of the process output for controller performance analysis.

7.4.1 Univariate system

For the univariate case, $m = 1$ and $l = 1$. Consider the process has a time delay of 'd' samples. Therefore, we have

$$L_u \;=\; \begin{bmatrix} 0 & 0 & \ldots & 0 \\ \ldots & \ldots & \ldots & \\ 0 & 0\ldots & 0 & \\ g_d & g_{d+1} & 0 & \ldots \\ \ldots & \ldots & \ldots & \ldots \end{bmatrix} \tag{7.67}$$

$$I - L_u L_u^{\dagger} \;=\; \begin{bmatrix} I_d & 0_{d\times(N-d)} \\ 0_{(N-d)\times d} & 0_{(N-d)\times(N-d)} \end{bmatrix} \tag{7.68}$$

$$(I - L_u L_u^{\dagger}) l_h \;=\; \begin{bmatrix} l_0 & l_1 & \ldots & l_{d-1} & 0 & \ldots & 0 \end{bmatrix}^T \tag{7.69}$$

where l_i represents the univariate impulse response coefficient of i-th lag and $l_h = \begin{bmatrix} l_0 & l_1 & \ldots \end{bmatrix}^T$ is a vector of the noise model impulse response coefficients. From the above equations we can see that *a priori* knowledge of L_u is not necessary for obtaining the MVC-benchmark variance for the univariate case. Only the first

'd' impulse response coefficients of the noise model are required. However,

$$L_u^\dagger = \begin{bmatrix} 0_{N\times(d-1)} & P_{N\times(N-d)} \end{bmatrix} \tag{7.70}$$

Therefore the first 'd' impulse response coefficients of the noise model are controller invariant, which means that the first 'd' impulse response coefficients of the closed loop noise model are same as those of the open loop noise model, and therefore can be identified from the routine operating data. Hence process time delay, 'd', is the only *a priori* knowledge required for the univariate case.

7.5 Multivariate minimum variance benchmark for the feedforward plus feedback control case

7.5.1 feedforward plus feedback minimum variance controller

Consider the case when some of the process disturbances are measurable and available for feedforward control. Measured disturbances are those process inputs which cannot be manipulated but affect process outputs. The state space representation can be modified to include the h-measured disturbances v_k as inputs:

$$x_{k+1} = Ax_k + \begin{bmatrix} B & B_v \end{bmatrix} \begin{bmatrix} u_k \\ v_k \end{bmatrix} + K^f e_k \tag{7.71}$$

$$y_k = Cx_k + D_v v_k + e_k \tag{7.72}$$

Equations (2.10)-(2.11) change to

$$y_f = \Gamma_N x_{t+1} + H_N u_f + H_N^v v_f + H_N^s e_f \tag{7.73}$$

$$= L_w^b w_p^b + L_u u_f + L_v v_f + L_e e_f \tag{7.74}$$

where

$$w_p^b = \begin{bmatrix} y_p \\ u_p \\ v_p \end{bmatrix}; \quad v_f = \begin{bmatrix} v_{t+1} \\ \dots \\ v_{t+N} \end{bmatrix}; \quad v_p = \begin{bmatrix} v_{t-N+1} \\ \dots \\ v_t \end{bmatrix}; \tag{7.75}$$

$$H_N^v = \begin{bmatrix} D_v & 0 & ... & 0 \\ CB_v & D_v & ... & 0 \\ ... & ... & ... & ... \\ CA^{N-2}B_v & CA^{N-3}B_v & ... & D_v \end{bmatrix} \tag{7.76}$$

The subspace matrices L_u, L_v and L_e can be identified from the closed loop data as explained in chapter 3. The minimum variance control law, as $N \longrightarrow \infty$, for feedback plus feed forward control can be obtained as,

$$u_f = -L_u^{\dagger}[L_w^b w_p^b] \tag{7.77}$$

7.5.2 MVC-benchmark

Similar to the previous section, we can derive

$$y_{t+1} = L_{y_p} \begin{bmatrix} y_{t-N+1} \\ ... \\ y_t \end{bmatrix} + L_{u_p} \begin{bmatrix} u_{t-N+1} \\ ... \\ u_t \end{bmatrix} + L_{v_p} \begin{bmatrix} v_{t-N+1} \\ ... \\ v_t \end{bmatrix} + L_0 \; e_{t+1} \tag{7.78}$$

where L_{y_p} and L_{u_p} are as defined before, and

$$L_{v_p} = L_w(1:m, (l+m)N+1:(l+m+h)N) \tag{7.79}$$

Similar to the previous section, define

$$L_{G^v} = \begin{bmatrix} G_0^v & G_1^v & ... & G_{N-1}^v & G_N^v \\ G_1^v & G_2^v & ... & G_N^v & 0 \\ ... & ... & ... & ... & ... \\ G_{N-1}^v & 0 & 0 & ... & 0 \end{bmatrix}; \quad \tilde{L}_v = \begin{bmatrix} 0 & 0 & ... & 0 \\ G_0^v & 0 & ... & 0 \\ ... & ... & ... & ... \\ G_{N-2}^v & G_{N-3}^v & ... & 0 \end{bmatrix};$$

$$\tilde{v}_p = \begin{bmatrix} v_{t+1} \\ v_t \\ ... \\ v_{t-N+1} \end{bmatrix}; \quad \tilde{v}_f = \begin{bmatrix} v_{t+2} \\ v_{t+3} \\ ... \\ v_{t+N+1} \end{bmatrix}$$

where G_i^v is the i-th impulse response coefficient (Markov parameter for multivariate systems) of the disturbance model corresponding to v_t. L_{G^v} can be formed from the subspace matrix L_v.

Equation (7.77) can be written as

$$u_f = -L_u^\dagger \{L_g u_p + L_{G^v} \tilde{v}_p + L_H \tilde{e}_p\} \tag{7.80}$$

The closed loop form of equation (7.74) can be written as

$$y_f = -(I - L_u L_u^\dagger)\{L_G \tilde{u}_p + L_{G^v} \tilde{v}_p + L_H \tilde{e}_p\} + \tilde{L}_v \tilde{v}_f + \tilde{L}_e \tilde{e}_f \tag{7.81}$$

Consider the measured and unmeasured disturbances enter the process at time = $t+1$, i.e.,

$$\begin{aligned} u_t &= u_{t-1} = \ldots = u_{t-N+1} = 0 \\ v_t &= v_{t-1} = \ldots = v_{t-N+1} = 0 \\ e_t &= e_{t-1} = \ldots = e_{t-N+1} = 0 \\ v_{t+2} &= v_{t+3} = \ldots = v_{t+N} = 0 \\ e_{t+2} &= e_{t+3} = \ldots = e_{t+N} = 0 \end{aligned}$$

Therefore we obtain the simplified expression

$$y_f = (I - L_u L_u^\dagger) L_{g^v} \; v_{t+1} + (I - L_u L_u^\dagger) L_h \; e_{t+1} \tag{7.82}$$

$$= \begin{bmatrix} \phi_0 \\ \ldots \\ \phi_{d-1} \\ 0 \\ \ldots \end{bmatrix} v_{t+1} + \begin{bmatrix} \psi_0 \\ \ldots \\ \psi_{d-1} \\ 0 \\ \ldots \end{bmatrix} e_{t+1} \tag{7.83}$$

$$= \Phi v_{t+1} + \Psi e_{t+1} \tag{7.84}$$

where L_h and Ψ are defined previously, $L_{g^v} = \begin{bmatrix} G_0^v \\ \ldots \\ G_N^v \end{bmatrix}$ and ϕ_i represents the Markov parameter of i-th lag of the closed loop noise model between v_t and y_t, if a minimum

variance controller were implemented on the system. It can also be shown that Φ can be identified from the closed-loop disturbances models. Therefore $L_{g^v}^{CL} = \begin{bmatrix} G_0^{v,CL} \\ \dots \\ G_N^{v,CL} \end{bmatrix}$ and $L_h^{CL} = \begin{bmatrix} L_0^{CL} \\ \dots \\ L_N^{CL} \end{bmatrix}$ can be used in the place of L_{g^v} and L_h respectively, where $L_{g^v}^{CL} = L_{yv}^{CL}(:, 1:h)$ and $L_h^{CL} = L_{ye}^{CL}(:, 1:m)$.

Therefore, the minimum variance control variance expression for the process output becomes

$$var[y_t]_{MVC} = \sum_{i=0}^{d-1} \{\phi_i var[v_t]\phi_i^T + \psi_i var[e_t]\psi_i^T\} \tag{7.85}$$

Similar to the previous section the first 'd' Markov parameters corresponding to the measured and unmeasured disturbances are controller invariant. Hence the closed loop measured and unmeasured disturbance subspace matrices L_v^{CL} and L_e^{CL} (see chapter 3) can be used in place of the open loop measured and unmeasured disturbance subspace matrices L_v and L_e in deriving the feedforward plus feedback multivariate MVC-benchmark. Hence only the subspace matrices L_u is required for obtaining the MVC-benchmark variance for the process output. The matrix L_u can be identified using the closed loop subspace identification method explained in chapter 3.

7.6 Algorithm for obtaining the MVC benchmark variance from routine operating data for multivariate systems

1. Obtain the deterministic subspace matrix L_u either from the open-loop data (see chapter 2) or from the closed-loop data (see section 3).

2. Collect routine closed loop operating data. By stochastic subspace identification for the case of measured disturbances estimate the vector of Markov parameters of

the closed loop noise model for both measured and stochastic disturbances, $L_{g^v}^{CL}$ and L_h^{CL} respectively, along with the noise variances $var[v_t]$ and $var[e_t]$.

3. Calculate the matrices $\Phi = (I - L_u L_u^{\dagger}) L_{g^v}^{CL}$ and $\Psi = (I - L_u L_u^{\dagger}) L_h^{CL}$.

4. Calculate $var[y_t]_{MVC}$ as

$$var[y_t]_{MVC} = \Phi \; var[v_t] \; \Phi^T + \Psi \; var[e_t] \; \Psi^T$$

5. Calculate the true process output variance $var[y_t]$ and compare it with the minimum variance benchmark $var[y_t]_{MVC}$.

7.7 Numerical example

Using a numerical example we demonstrate the equivalence of multivariate MVC-benchmark variance obtained using the subspace approach proposed in this chapter to that obtained through an interactor matrix filtering method presented in [42].

Consider the following 2×2 system (taken from [42]):

$$\mathbf{y_t} = G(z^{-1})\mathbf{u_t} + H(z^{-1})\mathbf{a_t}$$

where

$$\mathbf{y_t} = \begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix}; \quad \mathbf{u_t} = \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix}; \quad \mathbf{a_t} = \begin{bmatrix} a_1(t) \\ a_2(t) \end{bmatrix}$$

$$G(z^{-1}) = \begin{bmatrix} \frac{z^{-1}}{1-0.4z^{-1}} & \frac{z^{-2}}{1-0.1z^{-1}} \\ \frac{0.3z^{-1}}{1-0.1z^{-1}} & \frac{z^{-2}}{1-0.8z^{-1}} \end{bmatrix}; \quad H(z^{-1}) = \begin{bmatrix} \frac{1}{1-0.5z^{-1}} & \frac{-0.6}{1-0.5z^{-1}} \\ \frac{0.5}{1-0.5z^{-1}} & \frac{1}{1-0.5z^{-1}} \end{bmatrix}$$

The matrix L_u is constructed from the deterministic Markov parameters. The

intermediate matrices can be calculated for $N = 3$ as

$$L_u = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0.3 & 0 & 0 & 0 & 0 & 0 \\ 0.4 & 1 & 1 & 0 & 0 & 0 \\ 0.03 & 1 & 0.3 & 0 & 0 & 0 \end{bmatrix}; \quad L_e = \begin{bmatrix} 1 & -0.6 & 0 & 0 & 0 & 0 \\ 0.5 & 1 & 0 & 0 & 0 & 0 \\ 0.5 & -0.3 & 1 & -0.6 & 0 & 0 \\ 0.25 & 0.5 & 0.5 & 1 & 0 & 0 \\ 0.25 & -0.15 & 0.5 & -0.3 & 1 & -0.6 \\ 0.125 & 0.25 & 0.25 & 0.5 & 0.5 & 1 \end{bmatrix}$$

Now,

$$I - L_u L_u^\dagger = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.0826 & -0.2752 & 0 & 0 \\ 0 & 0 & -0.2752 & 0.9174 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

We check to see if $(I - L_u L_u^\dagger)$ is an interactor matrix by proving

$$\left[I - L_u L_u^\dagger\right] \begin{bmatrix} G_1 & 0 & \dots & 0 \\ G_2 & G_1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ G_d & G_{d-1} & \dots & G_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0.3 & 0 & 0 & \dots & 0 \\ 0.0248 & -0.1927 & 0 & \dots & 0 \\ -0.0826 & 0.6422 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \end{bmatrix} \quad (7.86)$$

with

$$K = \begin{bmatrix} 1 & 0 \\ 0.3 & 0 \end{bmatrix} + \begin{bmatrix} 0.0248 & -0.1927 \\ -0.0826 & 0.6422 \end{bmatrix} = \begin{bmatrix} 1.0248 & -0.1927 \\ 0.2174 & 0.6422 \end{bmatrix} \quad (7.87)$$

which is a full rank matrix.

We can obtain the Markov parameters of the noise model if a minimum variance

controller were implemented on the system as

$$\Psi = (I - L_u L_u^{\dagger}) L_h = \begin{bmatrix} 1 & -0.6 \\ 0.5 & 1 \\ -0.0275 & -0.1624 \\ 0.0917 & 0.5413 \\ 0 & 0 \\ \cdots & \cdots \end{bmatrix} \tag{7.88}$$

We can see that closed loop noise model Markov parameters under MVC matches with the result obtained in by Huang et al in [50] and 'Ko and Edgar' in [68].

7.8 Conclusions

Calculation of the multivariate performance index without using the interactor matrix is an important step toward practical application of multivariate performance assessment technique in addition to the advantage of reduced model structure error. It is shown in this chapter the design of the multivariate minimum variance controller can be done using subspace matrices. Using the subspace matrices the MVC-benchmark variance for the process outputs is obtained from closed loop data without having to first calculate the unitary interactor matrix. The method is expanded to the case of feedforward plus feedback control performance assessment. The equivalence of the subspace approach to the conventional transfer function approaching obtaining the MVC-benchmark variance is also proved. A numerical example is provided to illustrate the main results of the chapter.

Chapter 8

Dynamic multivariate analysis of variance

[1]

8.1 Introduction

The first step in controller performance analysis is comparing the process output variance with the theoretically achievable minimum output variance, the minimum variance benchmark. If the achieved controller performance is unsatisfactory but is close to the minimum variance control, then it is necessary to explore the feedforward control strategy [34, 60]. If feedforward control can indeed significantly reduce the output variance, then the remaining task is to select the measured disturbance variables for the implementation of feedforward control. In other words, we need to select those measured disturbance variables which contribute the most to the process output variance. The difficulty however is that measured disturbances may not be mutually independent. To solve this problem, we propose that the

[1] A version of this chapter was a part of the presentation made at the following conference
R. Kadali, B. Huang, and E. Tamayo. A case study on performance analysis and trouble shooting of an industrial model predictive control system. Proceedings of *American Control Conf., San Diego*, 1999.

measured disturbances can be de-coupled using principal component analysis, as will be discussed in this chapter.

It should be noted that the assumption of independence between measured and unmeasured disturbances in the analysis of variance is required and is actually guaranteed through the modeling procedure shown in the following sections. Any portion of unmeasured disturbance that is correlated with the measured disturbances is in fact "measurable" disturbance and can always be lumped together with the measured disturbance. The unmeasured disturbance produced in the modeling procedure is known as innovation sequence and is independent of the measured disturbances. This has also been confirmed through the data analysis.

8.2 Analysis of variance for feedforward control

For multivariate feedforward plus feedback control systems, closed loop response to both measured and unmeasured disturbances can be written in the transfer function representation as

$$y_t = G_e e_t + G_v v_t \tag{8.1}$$

where G_e and G_v are rational and proper transfer function matrices. y_t is the process output vector, e_t is white noise representing the driving force of the unmeasured disturbances, and v_t, which is not necessarily white noise, represents the measured disturbances. G_e and G_v can be estimated with any standard system identification tools. The variance of measured disturbances $E[v_t v_t^T] = \Sigma_v$ is decomposed as

$$E[v_t v_t^T] = \Sigma_v = L \Sigma_\delta L^T \tag{8.2}$$

where Σ_δ is a diagonal variance matrix. v_t is then mapped to δ_t by a linear transformation

$$\delta_t = L^T v_t \tag{8.3}$$

where $E[\delta_t \delta_t^T] = \Sigma_\delta$. This makes the elements of δ_t mutually independent.

Each element of $\delta_t(t)$ is then pre-whitened using univariate time-series analysis to obtain the corresponding element of the white noise driving force a_t. i.e., $\delta_t = G_a a_t$, where G_a is a diagonal transfer function matrix and $\delta_t = \begin{bmatrix} \delta_1(t) \\ ... \\ \delta_h(t) \end{bmatrix}$, $a_t = \begin{bmatrix} a_1(t) \\ ... \\ a_h(t) \end{bmatrix}$, where h is the number of the measure disturbance variables. With these transformations, equation (8.1) can be expanded as

$$\begin{aligned} y_t &= G_e e_t + G_v v_t \\ &= G_e e_t + G_v L \delta_t \\ &= G_e e_t + G_v L G_a a_t \end{aligned} \tag{8.4}$$

Multiplying the above equation with $z^{-d}D$, we get

$$z^{-d}Dy_t = z^{-d}DG_e e_t + z^{-d}DG_v L G_a a_t \tag{8.5}$$

where D is the unitary interactor matrix and z^{-1} is the back-shift operator. d is the order of the interactor matrix (the smallest integer that makes $z^{-d}D$ causal or the largest power of z in D [42]). This gives

$$\tilde{y}_t = \tilde{G}_e e_t + \tilde{G}_\delta a_t \tag{8.6}$$

$\tilde{G}_\delta a_t$ can be further expanded into individual components to give

$$\tilde{y}_t = \tilde{G}_e e_t + \tilde{G}_{\delta 1} a_{1t} + \tilde{G}_{\delta 2} a_{2t} + ... + \tilde{G}_{\delta h} a_{ht} \tag{8.7}$$

Therefore,

$$tr[Var(y_t)]_{MVC} = tr[Var(\tilde{y}_t)]_{MVC} \tag{8.8}$$

$$\begin{aligned} &= tr[Var(\tilde{G}_e e_t)]_{MVC} + tr[Var(\tilde{G}_{\delta 1} a_{1t})]_{MVC} \\ &\quad + tr[Var(\tilde{G}_{\delta 2} a_{2t})]_{MVC} + ... + tr[Var(\tilde{G}_{\delta h} a_{ht})]_{MVC} \end{aligned} \tag{8.9}$$

where

$$tr[Var(\tilde{G}_{\delta i} a_{it})]_{MVC} = trace\left\{ \sum_{k=0}^{d} [\tilde{G}_{\delta i}(k)][var(a_{it})][\tilde{G}_{\delta i}(k)]^T \right\} \tag{8.10}$$

and $\tilde{G}_{\delta i}(k)$ is the Markov parameter of k-th lag for the transfer functio nmatrix $\tilde{G}_{\delta i}$. $tr[Var(\tilde{G}_{\delta i}a_{it})]_{MVC}$ is the contribution to the output minimum-variance by the i^{th} principal component of the measured disturbances.

The analysis of variance provides the variance distribution among unmeasured disturbances, e_t, and the transformed measurable disturbances (or principal components), $a_t = \begin{bmatrix} a_1(t) \\ ... \\ a_h(t) \end{bmatrix}$. In order to implement the feedforward control, we have to trace back to the original measured disturbances from the principal components. To find the correlation between δ_t and the original measured disturbances v_t, we note that

$$v_t = L\delta_t \tag{8.11}$$

Therefore

$$E[v_t\delta_t^T] = L\Sigma_\delta \tag{8.12}$$

This yields

$$Corr(v_t, \delta_t) = \tilde{\Sigma}_v^{-\frac{1}{2}} E[v_t\delta_t^T]\Sigma_\delta^{-\frac{1}{2}} = \tilde{\Sigma}_v^{-\frac{1}{2}} L\Sigma_\delta^{-\frac{1}{2}} = M \tag{8.13}$$

where $\tilde{\Sigma}_v = diag(\Sigma_v)$, and Σ_δ is a diagonal variance matrix. The above equation provides a correlation matrix between the transferred disturbances, δ_t, and the original measured disturbances, v_t.

The correlation matrix, M, is useful for tracing the effects of the original measured disturbances on process variance. From the principal component contributing the highest to the output variance, the important disturbance variables can be identified.

8.3 The subspace matrices approach

Subspace matrices based approach has been the common thread that combines the theoretical work presented in this thesis. The dynamic analysis of variance needs

to be extended to the case of subspace approach, which as stated earlier does not require the calculation of the interactor matrix. However, the knowledge of the first few process Markov parameters is implicitly required in the form of the input subspace matrix, L_u.

An equivalent expression of the equation (8.1) for the subspace representation is,

$$Y_f = L_w \begin{bmatrix} Y_p \\ V_p \end{bmatrix} + L_{ye}^{CL} E_f + L_{yv}^{CL} V_f \tag{8.14}$$

where the subspace matrices L_{ye}^{CL} and L_{yv}^{CL} contain the Markov parameters of the closed loop noise model between the process outputs and the unmeasured and measured disturbances respectively. Refer to chapter 3 in deriving the above expression. The subspace matrices $\{L_{ye}^{CL},\ L_{yv}^{CL}\}$ are equivalent to the transfer function matrices $\{G_e, G_v\}$, and can be identified by regression as shown in chapter 2. v_t is then mapped to δ_t.

The subspace matrix, L_a, between $\delta_t \longrightarrow a_t$ is identified as follows:

Data Hankel matrices Δ_p and Δ_f are formed (refer to equations (2.4)-(2.7)) for the variable δ_t. Subspace matrices representation for the variable δ_t can be written as

$$\Delta_f = L_{w\delta}\Delta_p + L_a A_f \tag{8.15}$$

$\hat{\Delta}_f$ is found by the orthogonal projection of the row space of Δ_f into the row space spanned by Δ_p as

$$\hat{\Delta}_f = \Delta_f/\Delta_p \quad \Longrightarrow \quad L_{w\delta} = \Delta_f \Delta_p^{\dagger} \tag{8.16}$$

The vector of white noise sequence $a_f = \begin{bmatrix} a_t & a_{t+1} & \ldots & a_{t+j-1} \end{bmatrix} = \hat{\Delta}_f(1:h,:) - \Delta_f(1:h,:)$ is then obtained. The data Hankel matrix A_f is then formed from the white noise sequence a_f. The subspace matrix L_a is then obtained by the regression

$$L_a = (\Delta_f - \hat{\Delta}_f)\ A_f^{\dagger} \tag{8.17}$$

Equation (8.14) can be expressed as

$$Y_f = L_w \begin{bmatrix} Y_p \\ V_p \end{bmatrix} + L_{ye}^{CL} E_f + L_{y\delta}^{CL} A_f \tag{8.18}$$

where $L_\delta^{CL} = L_{yv}^{CL} L_N L_a$ and L_N is a diagonal matrix, $L_N = \begin{bmatrix} L & 0 & 0 \\ 0 & \dots & 0 \\ 0 & 0 & L \end{bmatrix}$.

As explained in chapter 7, pre-multiplying the 'vector of Markov parameters' with the matrix $(I - L_u L_u^\dagger)$ is equivalent to pre-multiplying the transfer function matrices with the term '$z^{-d}D$'. Therefore we can write (see equation (7.82))

$$y_f = (I - L_u L_u^\dagger) L_h^{CL} \; e_t + (I - L_u L_u^\dagger) L_\delta^{CL} \; a_t \tag{8.19}$$

$$= \begin{bmatrix} \psi_0 \\ \dots \\ \psi_{d-1} \\ 0 \\ \dots \end{bmatrix} e_t + \begin{bmatrix} \xi_0 \\ \dots \\ \xi_{d-1} \\ 0 \\ \dots \end{bmatrix} a_t \tag{8.20}$$

$$= \begin{bmatrix} \psi_0 \\ \dots \\ \psi_{d-1} \end{bmatrix} e_t + \begin{bmatrix} \xi_0^1 \\ \dots \\ \xi_{d-1}^1 \end{bmatrix} a_{1t} + \begin{bmatrix} \xi_0^2 \\ \dots \\ \xi_{d-1}^2 \end{bmatrix} a_{2t} + \dots + \begin{bmatrix} \xi_0^h \\ \dots \\ \xi_{d-1}^h \end{bmatrix} a_{ht} \tag{8.21}$$

where $L_h^{CL} = L_{ye}^{CL}(:, 1:m)$ and $L_\delta^{CL} = L_{y\delta}^{CL}(:, 1:h)$. Therefore we can write

$$\begin{aligned} tr\left[Var(y_t)\right]_{MVC} = \; & tr\left[\sum_{j=0}^{d-1} \psi_j Var(e_t) \psi_j^T\right] + tr\left[\sum_{j=0}^{d-1} \xi_j^1 Var(a_{1t}) (\xi_j^1)^T\right] \\ & + tr\left[\sum_{j=0}^{d-1} \xi_j^2 Var(a_{2t}) (\xi_j^2)^T\right] + \dots \\ & + tr\left[\sum_{j=0}^{d-1} \xi_j^h Var(a_{ht}) (\xi_j^h)^T\right] \end{aligned} \tag{8.22}$$

where $tr\left[\sum_{j=0}^{d-1} \xi_j^i Var(a_{it}) (\xi_j^i)^T\right]$ is the contribution to the output minimum-variance by the i^{th} principal component of the measured disturbances. The rest of the analysis is same as that outlined for the transfer function approach.

8.4 Simulations

The proposed analysis of variance methodology is tested in simulations. The system example is taken from [46].

$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \begin{bmatrix} \frac{z^{-1}}{1-0.4z^{-1}} & \frac{z^{-2}}{1-0.1z^{-1}} \\ \frac{0.3z^{-1}}{1-0.1z^{-1}} & \frac{z^{-2}}{1-0.8z^{-1}} \end{bmatrix} \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix} + \begin{bmatrix} \frac{0.6z^{-1}}{1-0.4z^{-1}} & \frac{0.1z^{-1}}{1-0.4z^{-1}} & \frac{0.2z^{-1}}{1-0.3z^{-1}} \\ \frac{0.5z^{-1}}{1-0.3z^{-1}} & \frac{0.3z^{-1}}{1-0.2z^{-1}} & \frac{0.4z^{-1}}{1-0.1z^{-1}} \end{bmatrix} \begin{bmatrix} v_1(t) \\ v_2(t) \\ v_3(t) \end{bmatrix} + \begin{bmatrix} \frac{1}{1-0.5z^{-1}} & \frac{-0.6}{1-0.5z^{-1}} \\ \frac{0.5}{1-0.5z^{-1}} & \frac{1}{1-0.5z^{-1}} \end{bmatrix} \begin{bmatrix} e_1(t) \\ e_2(t) \end{bmatrix} \tag{8.23}$$

Open loop input/output data is obtained by exciting the open loop system using a designed 'RBS' signal of magnitude 1 for the inputs, $\mathbf{u_k}$ and random numbers of standard deviation 0.1 for the white noise sequences, $\mathbf{e_k}$, in MATLAB-*Simulink*. The measured disturbance signals are designed as follows

$$\begin{aligned} v_1(t) &= \frac{0.6}{1-0.4z^{-1}} a_1(t) \\ v_2(t) &= \frac{0.4}{1-0.6z^{-1}} a_2(t) \\ v_3(t) &= \frac{0.3}{1-0.7z^{-1}} (0.6v_1(t) + a_3(t)) \end{aligned}$$

Where $\begin{bmatrix} a_1(t) \\ a_2(t) \\ a_3(t) \end{bmatrix}$ are random noise signals of 0.1 standard deviation. Note that different 'seeds' were used in MATLAB for generating the random noise signals $\{a_1(t), a_2(t), a_3(t), e_1(t), e_2(t)\}$. Its clear from the above that the measured disturbance signal $v_3(t)$ is correlated with $v_1(t)$. Hence the measured disturbance $v_3(t)$ is redundant for the feedforward control design. We will check if the dynamic multivariate analysis of variance would detect the redundancy. The analysis is conducted using both subspace and the transfer function approaches. Identical results with some minor differences (arising form the numerical rounding up errors in MATLAB) are obtained for both the approaches.

Principle Component	δ_1	δ_2	δ_3
Variance Contribution	tr$[Var(\tilde{G}_{\delta 1}a_{1t})]$ 5.8068e-4	tr$[Var(\tilde{G}_{\delta 2}a_{2t})]$ 1.1104e-4	tr$[Var(\tilde{G}_{\delta 3}a_{3t})]$ 4.3912e-5
Percentage	78.94 %	15.09 %	5.97 %

Table 8.1: Contribution of the principle components to process output variance

	δ_{1t}	δ_{2t}	δ_{3t}
DV1	-0.9750	0.0215	-0.5173
DV2	0.0166	0.9996	-0.0133
DV3	0.2215	-0.0155	-0.8557

Table 8.2: Correlation matrix table

From table (8.1) we can see that the principle components 1 and 2 are contributing the highest to the process output variance. From the correlation matrix in table (8.2) we can see that for the principle components 1&2 the measured disturbances DV1 and DV2 have the highest coefficients respectively. Hence it can be concluded that only two of the measured disturbances (1 and 2) are sufficient for designing the feed-forward control.

8.5 Conclusions

Dynamic multivariate analysis of variance helps in the selection of important measured disturbance variables for feedforward control. The dynamic multivariate analysis of variance is solved both in the traditional input-output transfer function framework and the subspace matrices approach. The former requires the interactor matrix and the latter does not. Thus the subspace approach simplifies the notation and facilitates its applications.

Chapter 9

Synthesis and analysis of an industrial MPC application

9.1 Introduction

The goal of advanced control applications in industry is not only to achieve optimal control of the process variables but also to maximize the profit achievable from the process through the optimization of the overall operation of the process. Implementation of advanced control applications is an exercise of balancing these dual goals while achieving process stability. A multivariate advanced control application aids in reducing the interactions between the process variables and optimizing process operations. Since advanced control applications have more involved computations using mathematical models, a slow sampled advanced control application is typically built on a faster sampling multiple PID controller system on the process. The objective of this chapter is to provide a practical guide to the performance analysis of industrial MPC applications.

This chapter discusses the industrial implementation of a commercial MPC application and proposes an MPC relevant controller performance assessment tool extending some of the theoretical work presented in the previous chapters.

We show how subspace matrix based approach can be used to define an MPC-relevant benchmark for assessing the performance of advanced control applications in industry.

The MPC discussed in this chapter was successfully commissioned. It ran for about 3-4 months and then shut down due to flow measurement sensor problem. Nevertheless, the methodology presented in this chapter is valuable and applicable to performance assessment of model predictive controls.

9.2 Process description

The process being considered is a large settling vessel designed to recover the "cream" of aerated hydrocarbons from a slurry of hydrocarbons, water, fines and sand. The hydrocarbons float to the top of the vessel due to their lower density and is recovered in the top stream. The higher density sands settle at the bottom of the vessel and is sent away through the bottom stream. A third outlet stream exists along the outer periphery of the middle of the vessel.

The settling vessel is a large rake thickener, which facilitates the flotation of hydrocarbons and the settling of sand. The feed valve position and the feed density are the measured disturbances for feedforward control. The feed is introduced in the middle of the vessel, the aerated hydrocarbons float over a weir circling the top and sand settles to the bottom. A hot water wash stream is introduced to the settling vessel to aid in the separation of aerated hydrocarbons from sand. Rakes rotating at the bottom of the vessel push the settled sand towards a small cone from where it is pumped away. A diagram of the settling vessel is shown in figure 9.1. The middle stream may be recycled back to the settling vessel or sent to a downstream separation process.

The depth of the lighter hydrocarbon bed floating at the top of the settling vessel is controlled by the rate of middle stream flow rate. A sight glass on the side of the vessel allows the operator to see the interface between the hydrocarbon bed and

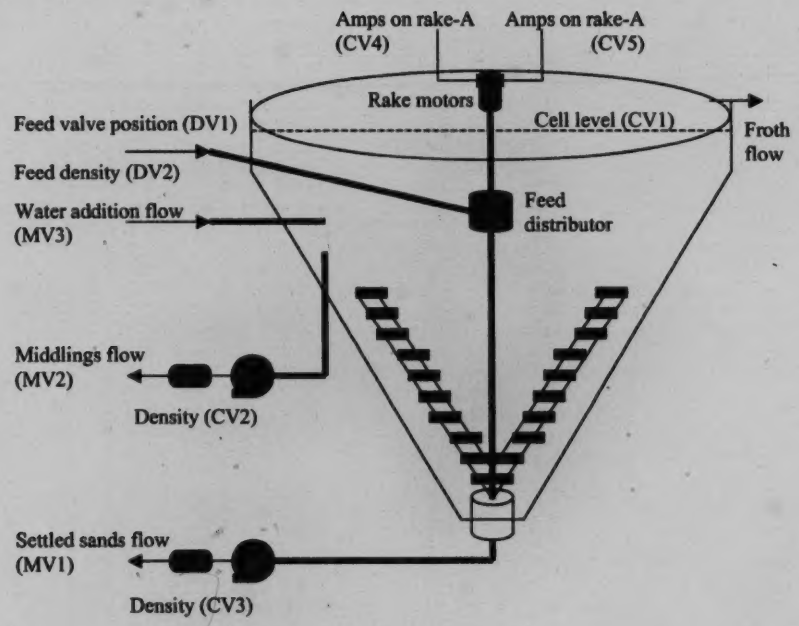


Figure 9.1: Settling vessel

the slurry. A variable speed pump provides control over the rate of middle stream withdrawal. The aerated hydrocarbons flow over the weir into a sloped launder, then flows by gravity to a Deaerator. The setpoint for level in the settling vessel is manipulated to maintain the interface level at a certain height and visible through the side sight glass. Under favorable operating conditions most of the hydrocarbons entering the settling vessel leave in the top stream. The recovery may drop when the quality of the oil sands feed deteriorates or when there is an upset in the settling vessel. Stable process operation is important to achieve high recoveries consistently from the settling vessel.

The following process variables are used in the advanced control application

Level of the cell CV1

Density of the middle flow stream CV2

Density of the bottom flow stream CV3

% Water in the top stream-flow CV4

Motor amperes on rake motor-A CVC1 (constraint)

Motor amperes on rake motor-B CVC2 (constraint)

Pump amperes on bottom stream flow CVC3 (constraint)

Setpoint for bottom stream flow pump speed controller MV1

Middle stream flow rate MV2

Setpoint for water wash stream flow pump speed controller MV3

Valve position on feed line to the settling vessel DV1

Density of feed to the settling vessel DV2

9.3 Identification

Open loop data with step changes to the manipulated variables is collected for the identification dynamic models for the process. Using DMI software, dynamic step response models for the process were identified and converted into transfer function models in the MPC-identification software. A MPC controller was then designed and tested in simulations and loaded into the DCS.

The following process transfer function models were identified for the system:

Variable	MV1	MV2	MV3
CV1	$\frac{-0.286}{45s^2+11s+1}$	$\frac{-0.73e^{-s}}{9s+1}$	$\frac{0.445(-10s+1)}{85s^2+15s+1}$
CV2	$\frac{-0.00045e^{-3s}}{34s^2+9s+1}$	$\frac{-0.0007(-11s+1)}{23s^2+10s+1}$	$\frac{-0.000142(57s+1)e^{-s}}{70s^2+12s+1}$
CV3	$\frac{-0.00188e^{-3s}}{10s+1}$	...	$\frac{0.0027(-10s+1)}{88s^2+15s+1}$
CV4	$\frac{-0.433e^{-7s}}{25s^2+8s+1}$	$\frac{-0.506e^{-4s}}{6s+1}$	...

Table 9.1: Identified transfer functions between the CVs and MVs.

Disturbance transfer function models for the system are identified as follows:

Variable	DV1	DV2
CV1	$\frac{2.86}{8s+1}$	...
CV2	$\frac{0.0074e^{-3s}}{7s+1}$	$\frac{0.404e^{-s}}{27s^2+9s+1}$
CV3	$\frac{0.0072}{65s^2+14s+1}$	$\frac{0.382(-6s+1)e^{-2s}}{40s^2+10s+1}$
CV4	$\frac{2.97e^{-2s}}{7.5s+1}$	...

Table 9.2: Identified transfer functions between the CVs and DVs.

9.4 Tuning and Optimization

The advanced control application is then tested online. Optimization function and weighting matrices for the controlled variables are carefully determined on the basis of the relative importance of the variables in maximizing the profitability of the process and achieving stability in the vessel operation. The optimization function chosen is maximization of the density difference between the bottom and middle streams, $max(CV3 - CV2)$. Similarly appropriate weighting matrices were chosen for the controlled variables. The process variables under the advanced control is plotted in figure (9.2).

9.5 Performance assessment

Assessing the performance of advanced controller applications is very different from the performance assessment of PID-type univariate controllers. The differences come from the fact that advanced controller applications are *multi-faceted*, in the sense that they are model-based and multivariate in nature, have an optimizer part in addition to the controller part and have application specific tuning parameters such as designed closed loop settling time.

Hence the performance assessment of an advanced controller application involves a more complex analysis of the different aspects of the application. We see some of the different methods of a *multi-faceted* analysis as follows.

9.5.1 Online predictions validation

Since the advanced control applications are model-based, validating the model predictions online in real-time is a critical step before the commissioning of the controller. Most advanced control applications have a bias updating term in the predictor to deal with small deviations of the models. By plotting the true value of a controlled variable with the model-predicted value the bias in the model predictions can be noticed. If significant bias is noticed the model needs to be re-identified. From figure (9.3) we can see that the model-predictions do not have any significant bias with the true values of the controlled variables. From the auto-correlations in figure (9.4) we can see that the prediction errors of the CVs are close to white noise.

9.5.2 Primary assessment of controller performance

Once an MPC application is installed and put on-line, the first step in the performance assessment of the application is analyzing the controller response to the (*a*) setpoint deviations of the controlled variables (feedback control) and (*b*) disturbances entering through the measured disturbance variables (feedforward control). Moving the manipulated variables in the right direction when the controlled variables deviate from their respective setpoints or when disturbances enter the process, is a critical indication of the performance of an advanced control application. If the controller moves any one of the manipulated variables in the wrong direction the process can spiral to unsteady operation.

Under closed loop, MVs are a function of setpoint deviations, (SP - CV), and the measured disturbances. From the process model we know the process gain relation between the MVs and CVs/DVs. For a setpoint deviation of controlled variables and / or a change in the measured disturbances in a particular direction (positive / negative) we expect the advanced controller to move the manipulated variables in a certain direction to narrow the setpoint deviation. By plotting the correlation coefficients between the 'setpoint deviations' and 'measured disturbances' with the 'manipulated variables' it can be verified if the controller is moving the manipulated

variables in the right direction.

From the transfer functions in the first row of table (9.1) we can see that when CV1 deviates below the setpoint (i.e., $(CV1_{SP} - CV1) < 0$), the controller should move $MV1$ $\uparrow$, $MV2$ $\uparrow$ and $MV3$ $\downarrow$. The designed MPC is doing exactly that in closed loop as can be seen from the cross-correlation coefficients plotted in figure (9.5). The process variable CV2 has a setpoint range instead of a single setpoint. Hence from the transfer functions in the second row of table (9.1) we can see that when CV2 deviates above the setpoint range (i.e., $CV2 > 1.4$), the controller should move $MV1$ $\uparrow$, $MV2$ $\uparrow$ and $MV3$ $\uparrow$ and vice-versa for $CV2 < 1.1$. Hence by plotting the cross-correlations coefficients between the CV2 setpoint deviations (i.e., $(1.4 - CV2)$ for $CV2 > 1.4$ and $(1.1 - CV2)$ for $CV1 < 1.1$) and MVs we can verify that the controller is moving the manipulated variables in the right direction as can be seen in figure (9.6). Similar observations can be made for CV3 and CV4 as shown in figures (9.7)-(9.8).

From (9.9)-(9.10) it is verified that the controller is moving the manipulated variables in the right direction for variation in the disturbance variables.

9.5.3 Controller optimizer performance assessment

The advanced control application on the settling vessel has two parts, the controller part and the optimizer part. The objective of the optimizer in the advance control application on the settling vessel is to maximize the density difference between the densities of the middle and bottom streams. When all the controlled variables are within their setpoint limits the optimizer part would take over and try to move the manipulated variables in the direction of maximizing the density difference, (CV3 - CV2). Hence the next step of performance analysis would be assessing the optimizer response when all the CVs are within their respective setpoint limits. From figure (9.11) we can clearly see that the MPC application is moving MV1 (the MV which has the highest effect on the process variables) in the right direction to maximize the density difference whenever both CV2 and CV3 are within the setpoint range.

9.5.4 Closed loop settling time based performance assessment

Certain advanced control applications, such as MPC, allow for setting the closed loop settling time as a tuning parameter. Hence the next step in the performance analysis of the controller is comparing the true closed loop settling time of the controlled variables with the designed settling time. Figure (9.12) shows the auto-correlation coefficients of the four CVs for the settling vessel. The vertical lines in each of the four plots represent the maximum closed-loop settling time specified in the controller design. From the figure we can see that the closed-loop settling times of the three controlled variables (CV2, CV3 and CV4) which determine the profitability from the settling process are well within the maximum closed-loop settling time specified in the controller design.

This concludes the initial performance analysis of the MPC application. From the analysis it can be concluded that the advanced control application is meeting all the control objectives in the optimization of the process performance on the settling vessel. The methods described so far enable us to assess the controller performance as 'good' or 'bad'.

The next step would be the benchmark-based controller performance assessment. The benchmark-based methods assess the 'optimality' of the controller performance with the benchmarks defining the optimal performance. These methods involve more rigorous analysis of the controller performance compared to the user defined benchmark for the optimal controller performance. The benchmark-based controller performance assessment techniques require partial knowledge of the process model for the multivariate case. It is shown in chapters 3, 5 and 7 that the required process knowledge can be estimated from the closed loop data if the open-loop process model is not available.

9.5.5 MVC-benchmark calculation using subspace approach with weighting matrices

The expressions for obtaining the MVC-benchmark for controller performance assessment have been derived in chapter 7. When certain weighting matrices for the controlled variables are included in the controller design then the expressions in chapter 7 have to be modified as follows.

Consider the case when the objective function for the minimum variance controller, as $N \longrightarrow \infty$, is

$$J = E\{ \sum_{k=1}^{N} [(r_{t+k} - y_{t+k})^T \; Q_k \; (r_{t+k} - y_{t+k})]\} \tag{9.1}$$

where $Q_k(m \times m)$ is a non-negative definite matrix. By letting $r_k = 0$ and using equation (7.74) we can write,

$$J = \min_{u_f^2} [\hat{y}_f^T \; Q \; \hat{y}_f] \tag{9.2}$$

$$= (L_w^b w_p^b + L_u u_f)^T \; Q \; (L_w^b w_p^b + L_u u_f) \tag{9.3}$$

where $Q = \begin{bmatrix} Q_1 & 0 & 0 \\ 0 & ... & 0 \\ 0 & ... & Q_N \end{bmatrix}$ has the weighting matrices along the diagonal blocks.

To obtain the minimum variance control law, we differentiate J with respect to u_f and set it to zero.

$$\frac{\partial J}{\partial u_f} = 2L_u^T Q L_w^b w_p^b + 2L_u^T Q L_u u_f = 0 \tag{9.4}$$

We obtain the minimum variance control law as:

$$u_f = -(L_u^T Q L_u)^\dagger L_u^T Q (L_w^b w_p^b) \tag{9.5}$$

Similar to equation 7.82, we can derive

$$y_f = \left[I - L_u (L_u^T Q L_u)^\dagger L_u^T Q\right] L_v \; v_{t+1} + \left[I - L_u (L_u^T Q L_u)^\dagger L_u^T Q\right] L_h \; e_{t+1} \tag{9.6}$$

$$= \begin{bmatrix} \phi_0^w \\ \cdots \\ \phi_{d-1}^w \\ 0 \\ \cdots \end{bmatrix} v_{t+1} + \begin{bmatrix} \psi_0^w \\ \cdots \\ \psi_{d-1}^w \\ 0 \\ \cdots \end{bmatrix} e_{t+1} = \Phi^w e_{t+1} + \Psi^w e_{t+1} \quad (9.7)$$

The minimum variance control variance expression for the process output is given by

$$Var[y_t]_{MVC} = \sum_{i=0}^{d-1} \phi_i^w Var[v_t](\phi_i^w)^T + \sum_{i=0}^{d-1} \psi_i^w Var[e_t](\psi_i^w)^T \quad (9.8)$$

$$= Var[y_t]_{MVC}^v + Var[y_t]_{MVC}^e \quad (9.9)$$

where $Var[y_t]_{MVC}^v$ and $Var[y_t]_{MVC}^e$ represent the contribution of measured and unmeasured disturbances respectively, to the process output variance under minimum variance control.

As shown in section 7.3 using theorem-1, the subspaces with the closed-loop noise models L_{yv}^{CL} and L_{ye}^{CL} (which can identified from the routine closed-loop data) can be used in the place of the open-loop noise models L_v and L_h in equation (9.6) to estimate the MVC-benchmark variance. Estimation of the closed loop subspace matrices from the routine operating data is an open-loop identification problem. Refer to chapter 2 for the methodology to estimate the subspace matrices. The subspace matrix L_u is constructed from the process Markov parameters formed from the process transfer function matrix shown in table 9.1.

The MVC-benchmark based performance index for the controller can therefore be defined as

$$\eta = \frac{trace\,(Var[y_t]_{MVC})}{trace\,(Var[y_t]_{actual})} = \frac{trace\,(Var[y_t]_{MVC}^v) + trace\,(Var[y_t]_{MVC}^e)}{trace\,(Var[y_t]_{actual})} \quad (9.10)$$

The MVC-benchmark based parameters are listed in table 9.3. As can be seen from table 9.3, the advanced control application on the settling process is giving 35.68% of the optimal performance based on the minimum variance control as benchmark.

parameter	Value
trace $(Var[y_t]^v_{MVC})$	5.3498
trace $(Var[y_t]^e_{MVC})$	2.3091
trace $(Var[y_t]_{MVC})$	7.6588
trace $(Var[y_t]_{actual})$	21.4671
η	0.3568
$1-\eta$	0.6432

Table 9.3: MVC-benchmark based controller performance assessment.

9.5.6 LQG-benchmark calculation using subspace approach with weighting matrices

Similar to the previous section, the expressions in chapter 5 have to be modified for the case of weighting matrices as follows.

The LQG objective function in this case can be written as

$$J = E\{ \sum_{k=1}^{N} [(y_{t+k} - r_{t+k})^T \; Q_k \; (y_{t+k} - r_{t+k}) \; + u_{t+k}^T \; R_k \; u_{t+k}] \; \} \quad (9.11)$$

where Q_k $(m \times m)$ and R_k $(l \times l)$ are non-negative definite weighting matrices and $N \longrightarrow \infty$. Consider the feedback plus feedforward controller case considered in section 5.4. By letting $r_k = 0$ we can write

$$J = \min_{u_f^2} \; [\hat{y}_f^T \; Q \; \hat{y}_f + u_f^T \; R \; u_f] \quad (9.12)$$

$$= (L_w^b w_p^b + L_u u_f)^T \; Q \; (L_w^b w_p^b + L_u u_f) + u_f^T \; R \; u_f \quad (9.13)$$

where $Q = \begin{bmatrix} Q_1 & 0 & 0 \\ 0 & ... & 0 \\ 0 & ... & Q_N \end{bmatrix}$ and $R = \begin{bmatrix} R_1 & 0 & 0 \\ 0 & ... & 0 \\ 0 & ... & R_N \end{bmatrix}$, have the weighting matrices along the diagonal blocks. We differentiate J with respect to u_f and set it to zero, to obtain the LQG control law as:

$$u_f = -(R + L_u^T Q L_u)^{-1} L_u^T Q L_w^b w_p^b \quad (9.14)$$

Equation 5.19 is modified as

$$K^w = (R + L_u^T Q L_u)^{-1} L_u^T Q;$$

(9.15)

Therefore we can derive

$$u_f = -K^w L_v v_{t+1} - K^w L_h e_{t+1} \tag{9.16}$$

$$= \begin{bmatrix} \omega_0^w \\ \omega_1^w \\ \cdots \\ \omega_{N-1}^w \end{bmatrix} v_{t+1} + \begin{bmatrix} \psi_0^w \\ \psi_1^w \\ \cdots \\ \psi_{N-1}^w \end{bmatrix} e_{t+1} \tag{9.17}$$

$$y_f = (I - L_u K^w) L_v v_{t+1} + (I - L_u K^w) L_h e_{t+1} \tag{9.18}$$

$$= \begin{bmatrix} \Upsilon_0^w \\ \Upsilon_1^w \\ \cdots \\ \Upsilon_{N-1}^w \end{bmatrix} v_{t+1} + \begin{bmatrix} \gamma_0^w \\ \gamma_1^w \\ \cdots \\ \gamma_{N-1}^w \end{bmatrix} e_{t+1} \tag{9.19}$$

we can calculate the LQG-benchmark variances of the process input and output as

$$Var[u_t]_{LQG} = \sum_{i=0}^{N-1} \omega_i^w \; Var[v_t] \; (\omega_i^w)^T + \sum_{i=0}^{N-1} \psi_i^w \; Var[e_t] \; (\psi_i^w)^T \tag{9.20}$$

$$= Var[u_t]^v_{LQG} + Var[u_t]^e_{LQG} \tag{9.21}$$

$$Var[y_t]_{LQG} = \sum_{i=0}^{N-1} \Upsilon_i^w \; Var[v_t] \; (\Upsilon_i^w)^T + \sum_{i=0}^{N-1} \gamma_i^w \; Var[e_t] \; (\gamma_i^w)^T \tag{9.22}$$

$$= Var[y_t]^v_{LQG} + Var[y_t]^e_{LQG} \tag{9.23}$$

As can be seen from the above expressions, the subspace matrices L_u, L_v and L_e are required for the calculation of the LQG-benchmark variances. The subspace matrix L_u is constructed from the process Markov parameters formed from the process transfer function matrix shown in table 9.1. Similarly the matrix L_v is constructed from the Markov parameters formed from the measured disturbances transfer function matrix shown in table 9.2. The subspace matrix L_e is identified from the open-loop data used in the identification of tables 9.1 and 9.2.

The benchmark variances are compared with the true variances of the process input and output. The performance indices for the LQG-benchmark based performance assessment are defined in section 5.5.

As can be seen from equations (9.16) and (9.18), the estimation of the LQG-benchmark requires the estimation of the open loop noise models of the measured and stochastic noise disturbances. For the advanced control application on the settling process, the open-loop noise model for measured disturbances is available from the identification step in section 9.3. The current stochastic noise model needs to be estimated from the closed-loop data used for controller performance analysis. For the advanced control application on the settling process, the model predictions for the controlled variables are archived along with the true values from the process. Using the predicted/true values of the controlled variables the open-loop stochastic noise model can be estimated. We can rewrite equation (5.36) as

$$Y_f = L_w^b W_p^b + L_u U_f + L_v V_f + L_e E_f \tag{9.24}$$

and

$$\hat{Y}_f = L_w^b W_p^b + L_u U_f + L_v V_f \tag{9.25}$$

Therefore the prediction error term can be extracted as

$$\Sigma_f = Y_f - \hat{Y}_f = L_e E_f \tag{9.26}$$

Using the least squares estimation method described in section 2.3 the subspace matrix L_e can be estimated. The subspace matrix L_e contains the Markov parameters of the stochastic noise model.

The LQG-benchmark based performance indices for the controller can therefore be defined as

$$\eta = \frac{trace\left(Var[y_t]_{LQG}\right)}{trace\left(Var[y_t]_{actual}\right)} = \frac{trace\left(Var[y_t]^v_{LQG}\right) + trace\left(Var[y_t]^e_{LQG}\right)}{trace\left(Var[y_t]_{actual}\right)} \tag{9.27}$$

$$\varepsilon = \frac{trace\left(Var[u_t]_{LQG}\right)}{trace\left(Var[u_t]_{actual}\right)} = \frac{trace\left(Var[u_t]^v_{LQG}\right) + trace\left(Var[u_t]^e_{LQG}\right)}{trace\left(Var[u_t]_{actual}\right)} \tag{9.28}$$

For the advanced control application on the settling process, The LQG-benchmark based parameters are listed in table 9.4.

The following can be deduced from table 9.4 about the performance of the advanced control application on the settling process based on the optimal LQG-control benchmark:

parameter	Value
trace $(Var[u_t]^v_{LQG})$	0.3202
trace $(Var[u_t]^e_{LQG})$	0.1074
trace $(Var[u_t]_{LQG})$	0.4276
trace $(Var[u_t]_{actual})$	3.4116
trace $(Var[y_t]^v_{LQG})$	4.2402
trace $(Var[y_t]^e_{LQG})$	2.9826
trace $(Var[y_t]_{LQG})$	7.2228
trace $(Var[y_t]_{actual})$	8.0919
η	0.8926
$1-\eta$	0.1074
ε	0.1253
$1-\varepsilon$	0.8747

Table 9.4: LQG-benchmark based controller performance assessment.

(a) In terms of the process output variance, the controller is giving 89.26% of the optimal performance.

(b) In terms of the process input variance, the controller is giving only 12.53% of the optimal performance.

Hence it can be concluded that the advanced control application is giving close to optimal performance with respect to the process output variance. However there is still some scope to reduce the process input variability to reduce the wear on the manipulated variables.

9.6 Conclusions

In this chapter the *multi-faceted* approaches for the performance assessment of industrial advanced control applications have been described and illustrated through an application on a settling vessel. It shows that the advanced control commissioned in this project achieved a relatively good performance. The proposed method for assessment of model predictive control can be applied to other industrial advanced

control systems.

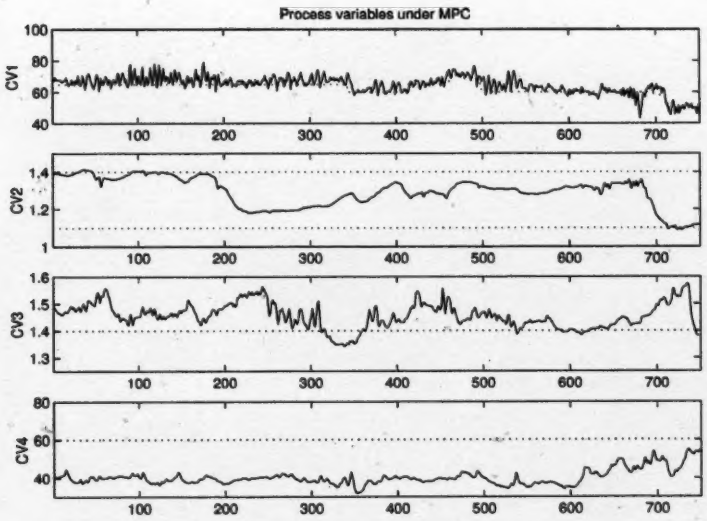


Figure 9.2: Closed loop data of the CVs under MPC.

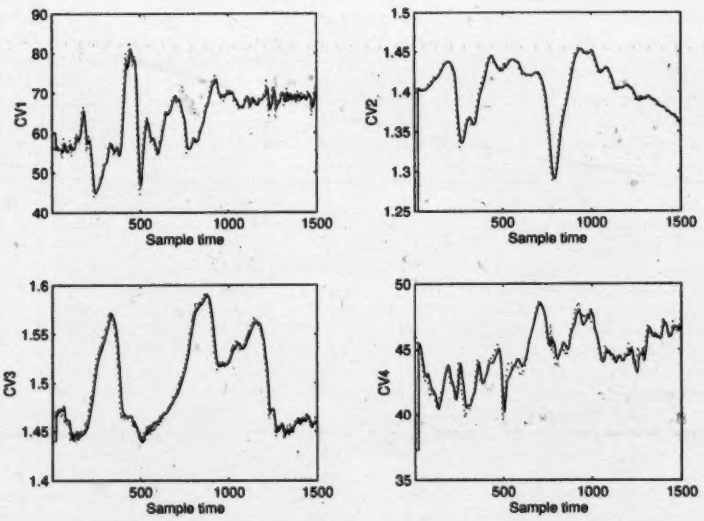


Figure 9.3: Comparing the model-predictions with the true values (–) of the controlled variables.

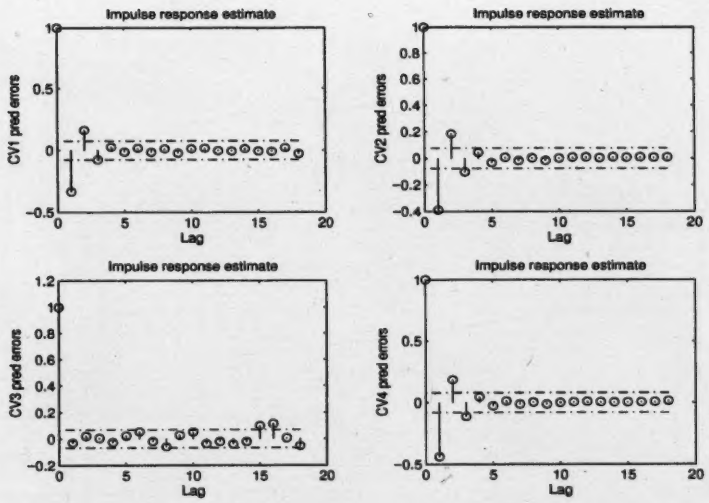


Figure 9.4: Auto correlation function of the prediction errors.

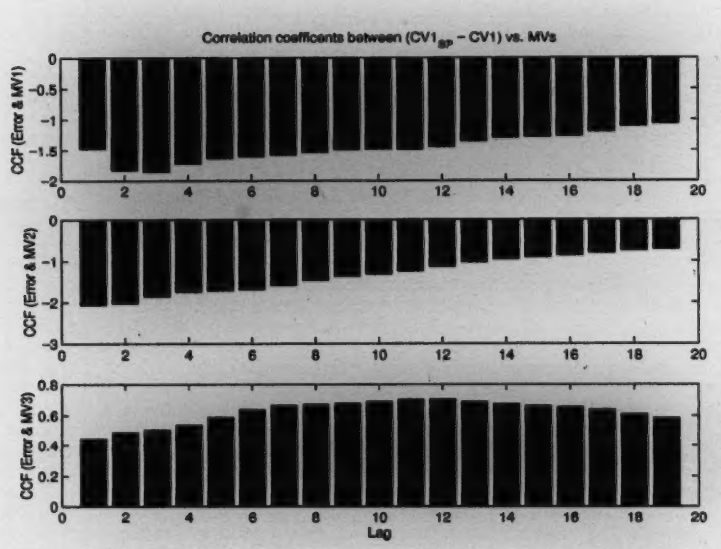


Figure 9.5: Cross correlation coefficients between $Error = (CV1_{SP} - CV1)$ and MVs.

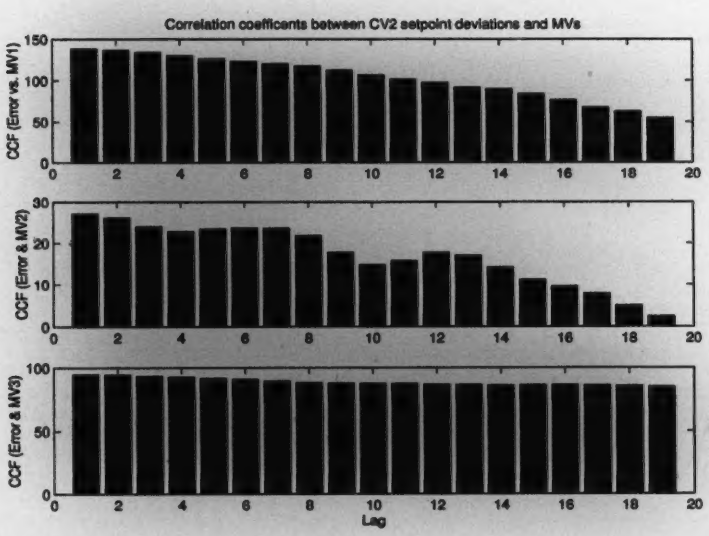


Figure 9.6: Cross correlation coefficients between $Error = (CV2_{SP} - CV2)$ and MVs.

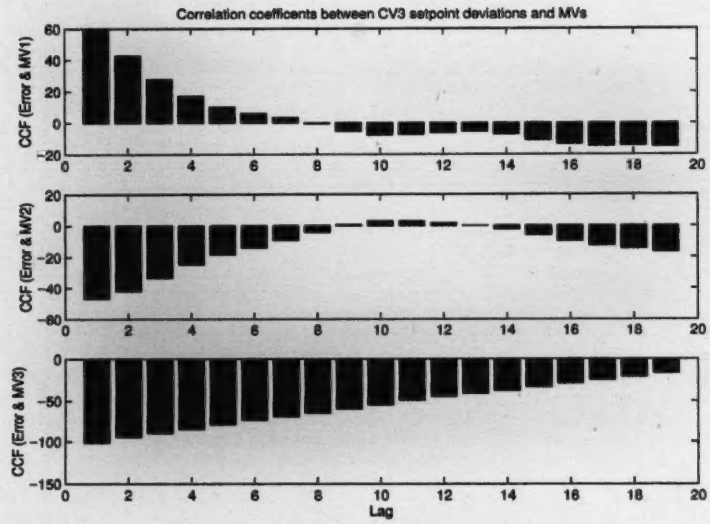


Figure 9.7: Cross correlation coefficients between $Error = (CV3_{SP} - CV3)$ and MVs.

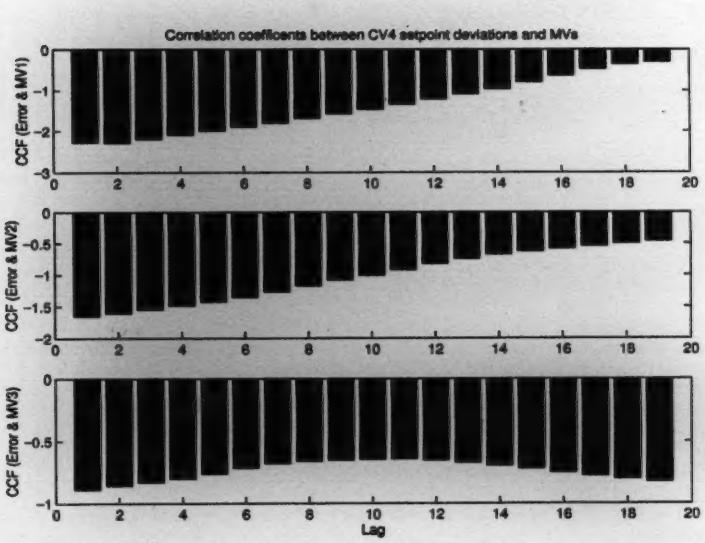


Figure 9.8: Cross correlation coefficients between $Error = (CV4_{SP} - CV4)$ and MVs.

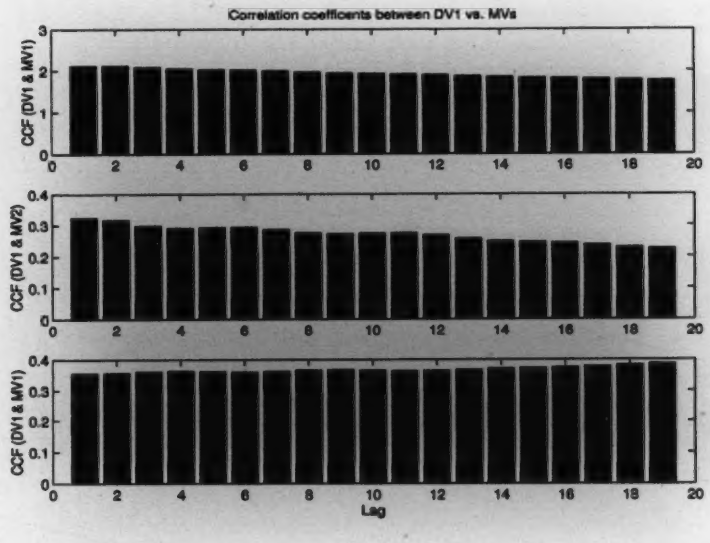


Figure 9.9: Cross correlation coefficients between $DV1$ and MVs.

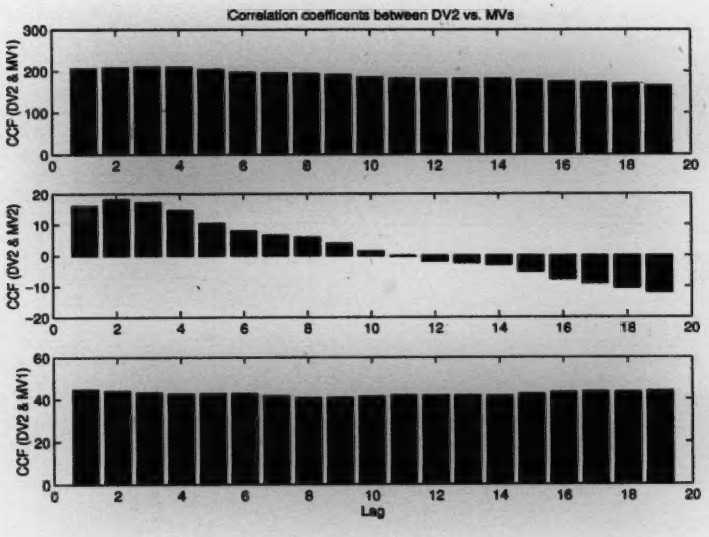


Figure 9.10: Cross correlation coefficients between $DV2$ and MVs.

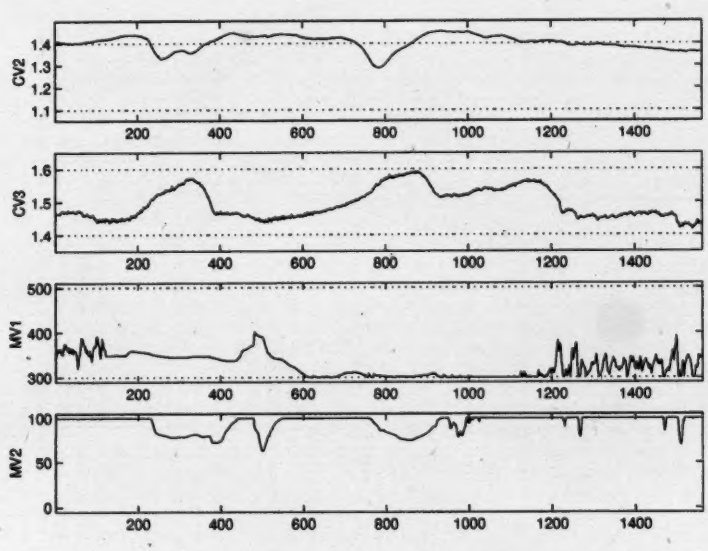


Figure 9.11: Optimization function operation.

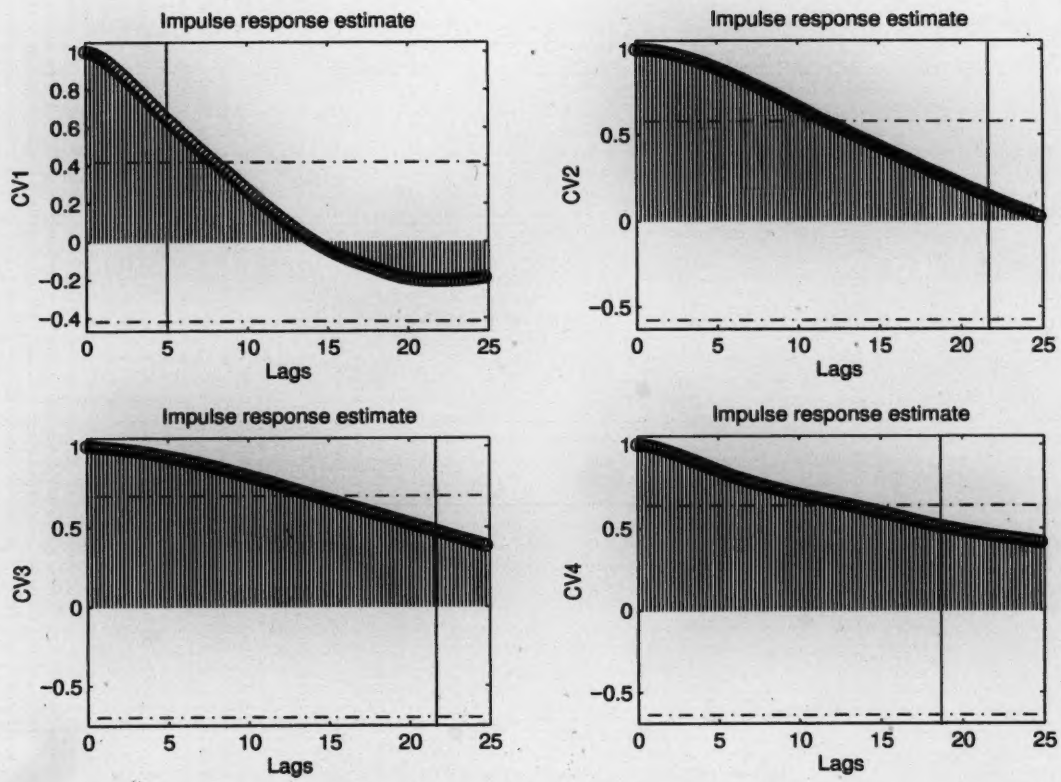


Figure 9.12: Auto correlation function of the CVs. The vertical lines represent the maximum closed loop settling times specified in the controller design.

Bibliography

[1] .. Special issue on statistical signal processing and control. *Automatica*, 30(1), 1994.

[2] .. Special issue on system identification. *Automatica*, 31(12), 1995.

[3] .. Special issue on subspace methods for system identification. *Signal Processing*, 52, 1996.

[4] B.D.O. Anderson and M.R. Gevers. Identifiability of linear stochastic systems operating under linear feedback. *Automatica*, 18:195–213, 1982.

[5] K.J. Astrom. Assessment of achievable performance of simple feedback loops. *International Journal of Adaptive Control and Signal Processing*, 5(1):3–19, 1991.

[6] K.J. Astrom and B. Wittenmark. *Computer Control Systems: Theory and Design.* Prentice-Hall, 1984.

[7] R.R. Bitmead, M. Gevers, and V. Wertz. *Adaptive Optimal Control: A thinking man's GPC.* Prentice Hall, 1990.

[8] S.P. Boyd and C.H. Barratt. *Linear Control Design.* Prentice Hall, Eaglewood Cliffs, New Jersey, 1991.

[9] J.W. Brewer. Kronecker products and matrix calculus in system theory. *IEEE trans. on Circuits and Systems*, 25(9):772–781, 1978.

[10] C.F. Camacho and C. Bordons. *Model Predictive Control.* Springer-Verlog London Limited, 1999.

[11] C.T. Chou and M. Verhaegen. Subspace algorithms for the identification of multivariable dynamic errors-in-variables models. *Automatica*, 33(10):1857–1869, 1997.

[12] D.W. Clarke, C. Mohtadi, and P.S. Tuffs. Generalized predictive control- part i. the basic algorithm. *Automatica*, 23(2):137–148, 1987.

[13] D.W. Clarke, C. Mohtadi, and P.S. Tuffs. Generalized predictive control- part ii. extensions and interpretations. *Automatica*, 23(2):149–160, 1987.

[14] C.R. Cutler, A. Morshedi, and J. Haydel. An industrial perspective on advanced control. In *AIChE Annual Meeting*, 1983.

[15] C.R. Cutler and B.L. Ramaker. Dynamic matrix control- a computer control algorithm. In *AIChE National Meeting*, 1979.

[16] C.R. Cutler and B.L. Ramaker. Dynamic matrix control- a computer control algorithm. In *Proceedings of the Joint Automatic Control Conference*, 1980.

[17] P.M.J. Van den Hof and R.J.P. Schrama. Identification and control—closed-loop issues. *Automatica*, 31(12):1751–1770, 1995.

[18] P.M.J. Van den Hof, R.J.P. Schrama, and O.H. Bosgra. An indirect method for transfer function estimation from closed-loop data. In *Proceedings of the 31st Conference on Decision and Control*, pages 1702 – 1706, Tucson, Arizona, December 1992.

[19] L. Desborough and T. Harris. Performance assessment measure for univariate feedback control. *Can. J. Chem. Eng.*, 70:1186–1197, 1992.

[20] L. Desborough and T.J. Harris. Performance assessment measures for univariate feedforward/feedback control. *Can. J. Chem. Eng.*, 71:605–616, 1993.

[21] L.D. Desborough and T.J. Harris. Control performance assessment. *Pulp and Paper Canada*, 94(11):441, 1994.

[22] P.G. Eriksson and A.J. Isaksson. Some aspects of control performance monitoring. In *Proc. 3rd IEEE conf. Control Applications*, pages 1029–1034, Glasgow, Scotland, 1994.

[23] W. Favoreel and B. De Moor. SPC: Subspace predictive control. *Tech. rep. 98-49, Katholieke Universiteit Leuven*, 1998.

[24] W. Favoreel, B. De Moor, M. Gevers, and P. van Overschee. Closed loop model-free subspace-based LQG-design. *Tech. rep. 98-108, Katholieke Universiteit Leuven*, 1998.

[25] W. Favoreel, B. De Moor, M. Gevers, and P. van Overschee. Model-free subspace based LQG-design. *Tech. rep. 98-34, Katholieke Universiteit Leuven*, 1998.

[26] W. Favoreel, B. De Moor, P. Van Overschee, and M. Gevers. Model-free subspace-based LQG-design. In *Proceedings of the American Control Conference*, pages 3372–3376, June 1999.

[27] U. Forssell and L. Ljung. Closed-loop identification revisited. *Automatica*, 35(7):1215–1241, 1999.

[28] J. Gao, R. Patwardhan, K. Akamatsu, Y. Hashimoto, G. Emoto, S.L. Shah, and B. Huang. Performance evaluation of two industrial mpc controllers. *Control Engineering Practice*, Accepted for publication, 2003.

[29] C. Garcia, D. Prett, and M. Morari. Model predictive control: Theory and praqctice - a survey. *Automatica*, 25(3):335–348, 1989.

[30] C.E. Garcia and A.M. Morshedi. Quadratic programming solution of dynamic matrix control (QDMC). *Chem. Eng. Commun.*, 46:73–86, 1986.

[31] A. Graham. *Kronecker Products and Matrix Calculus with Applications*. John Wiley & sons, 1981.

[32] I. Gustavsson, L. Ljung, and T. Soderstrom. Identification of processes in closed loop — identifiability and accuracy aspects. *Automatica*, 13:59–75, 1977.

[33] T. Hagglund. A control-loop performance monitor. *Control Engineering Practice*, 5(11):1543, 1995.

[34] T. Harris. Assessment of control performance. *Can. J. Chem. Eng.*, 67:856–861, 1989.

[35] T.J. Harris, F. Boudreau, and J.F. MacGregor. Performance assessment of multivariable feedback controllers. In *1995 AIChE Annual Meeting*, Miami Beach, FL, 1995.

[36] T.J. Harris, F. Boudreau, and J.F. MacGregor. Performance assessment of multivariable feedback controllers. *Automatica*, 32(11):1505–1518, 1996.

[37] T.J. Harris, C. Seppala, P. Jofriet, and B. Surgenor. Plant-wide feedback control performance assessment using an expert system framework. *Control Engineering Practice*, 4:1297, 1996.

[38] T.J. Harris, C.T. Seppala, and L.D. Desborough. A review of performance monitoring and assessment techniques for univariate and multivariate control systems. *J. of Process Control*, 9:1–17, 1999.

[39] D.A. Harville. *Matrix Algebra From a Statistician's Perspective.* Springer-Verlag, New York, 1997.

[40] H. Hjalmarsson, M. Gevers, and F. Bruyne. For model-based control design, closed-loop identification gives better performance. *Automatica*, 32(12):1659–1673, 1996.

[41] P.M.J. Van Den Hof and R.J.P. Schrama. An indirect method for transfer function estimation from closed loop data. *Automatica*, 29(6):1523–1527, 1993.

[42] B. Huang. *Multivariate Statistical Methods for Control Loop Performance Assessment.* PhD thesis, Department of Chemical Engineering, University of Alberta, Edmonton, Alberta, Canada, 1997.

[43] B. Huang and S.L. Shah. Closed-loop identification: a two-step approach. *Journal of Process Control*, 17(6), 1997.

[44] B. Huang and S.L. Shah. The role of the unitary interactor matrix in the explicit solution of the singular LQ output feedback control problem. *Automatica*, 33(11):2071–2075, 1997.

[45] B. Huang and S.L. Shah. Practical issues in multivariate control loop performance assessment. *Journal of Process Control*, 8(5-6):421–430, 1998.

[46] B. Huang and S.L. Shah. *Control Loop Performance Assessment: Theory and Applications*. Springer Verlag, London, 1999.

[47] B. Huang, S.L. Shah, and H. Fujii. Identification of the time delay/interactor matrix for MIMO systems using closed-loop data. In *Proc. 13th IFAC World Congress*, volume M, pages 355–360, San Francisco, July 1996.

[48] B. Huang, S.L. Shah, E.K. Kwok, and J. Zurcher. Performance assessment of multivariate control loops on a paper machine headbox. *Canadian Journal of Chemical Engineering*, 75(1):134–142, 1997.

[49] B. Huang, S.L. Shah, and K.Y. Kwok. Case studies of performance monitoring of multivariable processes. In *The 45th CSChE Conference*, Quebec, Canada, October 1995.

[50] B. Huang, S.L. Shah, and K.Y. Kwok. Good, bad or optimal? performance assessment of multivariable processes. *Automatica*, 33(6):1175–1183, 1997.

[51] B. Huang, S.L. Shah, and R. Miller. Feedforward plus feedback controller performance assessment of mimo systems. *IEEE Trans. on Control System Technology*, 8(3)(3):580–587, 2000.

[52] M. Jansson and B. Wahlberg. A linear regression approach to state-space subspace system indetification. *Signal Processing*, 52:103–129, 1996.

[53] M. Jansson and B. Wahlberg. On consistency of subspace methods for system identification. *Automatica*, 34(12):1507–1519, 1998.

[54] P. Jofriet, C. Seppala, M. Harvey, B. Surgenor, and T.J. Harris. An expert system for control loop performance. In *Pulp and Paper Canada*, page 207, 1996.

[55] R.A. Johnson and D.W. Wichern. *Applied Multivariate Statistical Analysis.* Prentice-Hall, 1982.

[56] R. Kadali and B. Huang. A data based subspace approach to predictive controller design. In *The 50th CSChE Conference, Montreal, Canada,* November 2000.

[57] R. Kadali and B. Huang. Controller performance assessment using LQG-benchmark obtained under closed loop. *ISA Transactions*, 2001. Accepted for publication.

[58] R. Kadali and B. Huang. Estimation of dynamic matrix and noise model for model predictive control using closed loop data. *Industrial and Engineering Chemistry Research*, 41(4):842–852, 2002.

[59] R. Kadali, B. Huang, and A.J. Rossiter. A data driven subspace approach to predictive controller design. *Control Engineering Practice*, 2002. Accepted for publication.

[60] R. Kadali, B. Huang, and E.C. Tamayo. A case study on performance analysis and troubling shooting of an industrial model predictive control system. In *Proceedings of 1999 American Control Conference*, pages 642–646, 1999.

[61] T. Kailath. *Linear System Theory.* Prentice Hall, Englewood Cliffs, NJ, 1980.

[62] L. C. Kammer. *Performance Monitoring and Assessment of Optimality under a Linear Quadratic Criterion.* PhD thesis, Department of Systems Engineering, The Australian National University, Canberra, Australia, 1998.

[63] L.C. Kammer, R.R. Bitmead, and P.L. Bartlett. Signal-based testing of lq-optimality of controllers. In *Proceedings of 1996 IFAC World Congress*, June-July 1996.

[64] L.C. Kammer, R.R. Bitmead, and P.L. Bartlett. Optimal controller properties from closed-loop experiments. *Automatica*, 34(1):83–91, 1998.

[65] S.J. Kendra and A. Cinar. Controller performance assessment by frequency domain techniques. *Journal of Process Control*, 7(3):181–194, 1997.

[66] P. Kesavan and J.H. Lee. Diagnostic tools for multivariable model-based control systems. *Ind. Eng. Chem. Res.*, 36:2725–2738, 1997.

[67] T. Knudsen. Consistency analysis of subspace identification methods based on linear regression approach. *Automatica*, 37:81–89, 2001.

[68] B.S. Ko and T.F. Edgar. Performance assessment of multivariate feedback control systems. *Automatica*, 37(5):899–905, 2001.

[69] D.J. Kozub. Controller performance monitoring and diagnosis: Experience and challenges. *AIChE Symposium Series*, 93(316):83, 1997.

[70] D.J. Kozub and C.E. Garcia. Monitoring and diagnosis of automated controllers in the chemical process industries. In *AIChE Annual Meeting*, St. Louis,MO, Nov. 9 1993.

[71] S. Lakshminarayanan, G. Emoto, S. Ebara, K. Tomida, and S.L. Shah. Closed loop identification and control loop reconfiguration: an industrial case study. *Journal of Process Control*, 11:587–599, 2001.

[72] W.E. Larimore. System identification, reduced-order filtering and modeling via canonical variate analysis. In H.S. Rao and T. Dorato, editors, *Proceedings of 1983 American Control Conference*, pages 445–451. IEEE, New York, 1983.

[73] W.E. Larimore. Statistical optimality and canonical variate analysis system identification. *Signal Processing*, 52:131–144, 1996.

[74] W.E. Larimore. Canonical variate analysis in control and signal processing. In T. Katayama and S.Sugimoto, editors, *Statistical Methods in Control and Signal Processing*, pages 83–120. Marcel Dekkar, New York, 1997.

[75] W.E. Larimore. Canonical variate analysis in identification, filtering, and adaptive control. In *Proceedings of the 29th Conference on Decision and Control*, pages 596–604, December 1990.

[76] W.E. Larimore. System identification of feedback and 'causality' structure using canonical variate analysis. In *Preprints 11th IFAC Symposium on System Identification*, pages 1101–1106, July 1997.

[77] W.E. Larimore. Optimal reduced rank modeling, prediction, monitoring, and control using canonical variate analysis. In *Preprints IFAC Advanced Control of Chemical Processes*, pages 61–66, June 1997.

[78] J.H. Lee and B. Cooley. Recent advances in model predictive control and other related areas. In *Chemical Process Control-CPC V, CACHE*, pages 201–216, 1996.

[79] A. Lindquist and G. Picci. Canonical correlation analysis, approximate covariance extension, and identification of stationary time series. *Automatica*, 32(5):709–733, 1996.

[80] L. Ljung. *System Identification: Theory for the user.* Prentice-Hall, Englewood, NJ, 2nd edition, 1987.

[81] L. Ljung and T. McKelvey. Subspace identification from closed loop data. *Signal Processing*, 52:209–215, 1996.

[82] C.B. Lynch and G.A. Dumont. Closed loop performance monitoring. In *Proc. 2nd IEEE Conf. Control Applications*, pages 835 – 840, Vancouver, B.C., Sept.13-16 1993.

[83] C.B. Lynch and G.A. Dumont. Control loop performance monitoring. *IEEE Trans. Control Sys. Tech*, 4(2):185–192, 1996.

[84] C.A.R. McNabb and J.S. Qin. MIMO control performance monitoring based on subspace projections. In *AIChE Annual Conference, Reno*, 2001.

[85] T. Miao and D.E. Seborg. A monitoring strategy for flow and pressure control loops. In *The 2nd Asia-Pacific Conf. on Control and Measurement*, Wuhan-Chongqing, China, June 1995.

[86] R. Miller and B. Huang. Perspectives on multivariate feedforward/feedback controller performance measures for process diagnosis. In *Proceedings of ADCHEM*, page 435, Banff, Canada, 1997.

[87] M. Moonen, B. De Moor, L. Vandenberghe, and J. Vandewalle. On- and off-line identification of linear state-space models. *International Journal of Control*, 49(1):219–232, 1989.

[88] M. Morari. chapter model predictive control: Multivariable control technique of choice in the 1990s? *Advances in Model-Based Predictive Control*, 1994.

[89] Y. Mutoh and R. Ortega. Interactor structure estimation for adaptive control of discrete-time multivariable nondecouplable systems. *Automatica*, 29(3):635–647, 1993.

[90] T.S. Ng, G.C. Goodwin, and B.D.O. Anderson. Identifiability of linear dynamic system operating in closed-loop. *Automatica*, 13:477–485, 1977.

[91] P. Van Overschee and B. De Moor. N4SID: Subspace algorithm for the identification of combined deterministic-stochastic systems. *Automatica*, 30(1):75–93, 1994.

[92] P. Van Overschee and B. De Moor. A unifying theorem for three subspace system identification algorithms. *Automatica*, 31(12):1877–1883, 1995.

[93] P. Van Overschee and B. De Moor. *Subspace Identification for Linear Systems: Theory implementation applications.* Kluwer Academic Publishers, 1996.

[94] P. Van Overschee and B. De Moor. Closed loop subspace system identification. In *Proceedings of the 36th Conference on Decision and Control*, pages 1848–1853, 1997.

[95] J.G. Owen, D. Read, H. Blekkenhorst, and A.A. Roche. A mill protype for automatic monitoring of control loop performance. In *Proc. Control Systems'96*, page 171, 1996.

[96] R. Patwardhan. *Studies in synthesis and analysis of model predictive controllers.* PhD thesis, Dept of chemical and materials engineering, University of Alberta, 1999.

[97] D.M. Prett and R.D. Morari. Optimization and constrained multivariable control of a catalytic cracking unit. In *Proceedings of the joint ACC*, 1980.

[98] S.J. Qin. Control performance monitoring - a review and assessment. *Computers and Chemical Engineering*, 23:173–186, 1998.

[99] S.J. Qin and T.A. Badgwell. An overview of industrial model predictive control technology. In *Proceedings of CPC-V*, Tahoe City, California, 1996.

[100] R.R. Rhinehart. A watch dog for controller performance monitoring. In *Proceedings of the 1995 American Control Conference*, pages 2239 – 2240, Seattle, Washington, U.S.A., 1995.

[101] J. Richalet, A. Rault, J.L. Testud, and J. Papon. Model predictive heauristic control: Applications to industrial processes. *Automatica*, 14:413–428, 1978.

[102] D.D. Ruscio. A method for identification of combined deterministic stochastic systems. In *Applications of Computer Aided Time Series Modeling. Editor: Aoki, M. and A. Hevenner*, pages 181–235. Springer-Verlag, 1997.

[103] D.D. Ruscio. Model based predictive control: An extended state space approach. In *Proceedings of the 36th Conference on Decision and Control*, pages 3210–3217, 1997.

[104] D.D. Ruscio. Model predictive control and identification: A linear state space model approach. In *Proceedings of the 36th Conference on Decision and Control*, pages 3202–3209, 1997.

[105] D.D. Ruscio. On subspace identification of the extended observability matrix. In *Proceedings of the 36th Conference on Decision and Control*, pages 1841–1847, 1997.

[106] D.D. Ruscio and B. Foss. On state space model based predictive control. In *IFAC Dynamics and Control of Process Systems*, 1998.

[107] C.D. Schaper, W.E. Larimore, D.E. Seborg, and D.A. Mellichamp. Identification of chemical processes using canonical variate analysis. *Computers and Chemical Engineering*, 18(1):55–69, 1994.

[108] R.J.P. Schrama. An open-loop solution to the approximate closed-loop approximation problem. In *Proceedings of IFAC identification and system parameter estimation*, pages 761–766, Budapest, Hungary, 1991.

[109] S.L. Shah, C. Mohtadi, and D.W. Clarke. Multivariable adaptive control without a prior knowledge of the delay matrix. *Systems and Control Letters*, 9:295–306, 1987.

[110] T. Soderstrom and P. Stoica. *System Identification.* Prentice Hall International, UK, 1989.

[111] N. Stanfelj, T.E. Marlin, and J.F. MacGregor. Monitoring and diagnosing process control performance: the single-loop case. *Ind. Eng. Chem. Res.*, 32:301–314, 1993.

[112] A. Stenman. *Model on Demand: Algorithm, Analysis and Applications.* PhD thesis, Dept of EE, Link ping University, SE-581 83 Link ping, Sweden, Apr 1999.

[113] A.K. Tangirala, S. Lakshminarayanan, and S.L. Shah. Closed-loop identification using canonical variate analysis. In *The 47th CSChE Conference, Edmonton, Canada*, November 1997.

[114] N.F. Thornhill, M. Oettinger, and P. Fedenczuk. Refinery-wide control loop performance assessment. *Journal of Process Control*, 9(2):109–124, 1999.

[115] N.F. Thornhill, R. Sadowski, R. Davis, J.R. Fedenczuk, P. Knight, M.J. Prichard, and D. Rothenberg. Practical experiences in refinery control loop performance assessment. In *UKACC'96*, 1996.

[116] M.L. Tyler and M. Morari. Performance assessment for unstable and nonminimum-phase systems. Technical Report IfA-Report No 95-03, California Institute of Technology, 1995.

[117] M.L. Tyler and M. Morari. Performance monitoring of control systems using likelihood methods. *Automatica*, 32:1145–1162, 1996.

[118] P.M.J. Van den Hof and R.J.P. Schrama. An indirect method for transfer function estimation from closed loop data. *Automatica*, 29(6):1523–1527, 1993.

[119] M. Verhaegen. Identification of the deterministic part of mimo state space models given in innovations form from input-output data. *Automatica*, 30(1):61–74, 1994.

[120] M. Verhaegen and P. Dewilde. Subspace model identification part 1. the output-error state-space model identification class of algorithms. *International Journal of Control*, 56(5):1187–1210, 1992.

[121] M. Verhaegen and P. Dewilde. Subspace model identification part 2. analysis of the elementary output-error state-space model identification algorithm. *International Journal of Control*, 56(5):1211–1241, 1992.

[122] Michel Verhaegen. Application of a subspace model identification technique to identify LTI systems operating on closed-loop. *Automatica*, 29(4):1027–1040, 1993.

[123] W.J. Vetter. Derivative operations on matrices. *IEEE trans. on* **AC**, pages 241–244, 1970.

[124] W.J. Vetter. Matrix calculus operations and taylor expansions. *SIAM review*, 15(2):352–369, 1973.

[125] A. Vishnubhotla, S.L. Shah, and B. Huang. Feedback and feedforward performance analysis of the shell industrial closed-loop data set. In *Proceedings of ADCHEM*, pages 295–300, Banff, Canada, 1997.

[126] Alexander Weinmann. *Uncertain Models and Robust Control.* Springer-Verlag, New York, 1991.

Appendix A

Closed loop subspace identification when a FF plus FB controller is acting on the process

Consider a feedback and feedforward controller acting on the process (2.1)-(2.2). The controller can be expressed in transfer function form as,

$$u_k = Q_1 \ (r_k - y_k) + Q_2 \ v_k$$

where v_k represents the measured disturbance variable. Assume that the measured disturbance variables are uncorrelated with the setpoint changes or the unmeasured disturbances. The controller subspace representation (3.2)-(3.3) changes to

$$\begin{aligned} x^c_{(k+1)} &= A_c x^c_k + \begin{pmatrix} B_c & B^f_c \end{pmatrix} \begin{pmatrix} r_k - y_k \\ v_k \end{pmatrix} \\ u_k &= C_c x^c_k + \begin{pmatrix} D_c & D^f_c \end{pmatrix} \begin{pmatrix} r_k - y_k \\ v_k \end{pmatrix} \end{aligned}$$

We derive

$$U_f = \Gamma^c_N X^c_f + H^c_N R_f - H^c_N Y_f + H^f_N V_f \qquad \text{(A.1)}$$

Therefore, we can write

$$\begin{aligned} U_f &= \Gamma_N^c X_f^c + H_N^c R_f - H_N^c(\Gamma_N X_f + H_N U_f + H_N^v V_f + H_N^s E_f) + H_N^f V_f \\ &= (I + H_N^c H_N)^{-1}\Gamma_N^c X_f^c - (I + H_N^c H_N)^{-1} H_N^c \Gamma_N X_f \\ &\quad +(I + H_N^c H_N)^{-1} H_N^c R_f - (I + H_N^c H_N)^{-1} H_N^c H_N^v V_f \\ &\quad -(I + H_N^c H_N)^{-1} H_N^c H_N^s E_f + (I + H_N^c H_N)^{-1} H_N^f V_f \end{aligned} \tag{A.2}$$

Similarly,

$$\begin{aligned} Y_f &= \Gamma_N X_f + H_N(\Gamma_N^c X_f^c + H_N^c R_f - H_N^c Y_f + H_N^f V_f) + H_N^v V_f + H_N^s E_f \\ &= (I + H_N H_N^c)^{-1}\Gamma_N X_f + (I + H_N H_N^c)^{-1} H_N \Gamma_N^c X_f^c \\ &\quad +(I + H_N H_N^c)^{-1} H_N H_N^c R_f + (I + H_N H_N^c)^{-1} H_N H_N^f V_f \\ &\quad +(I + H_N H_N^c)^{-1} H_N^v V_f + (I + H_N H_N^c)^{-1} H_N^s E_f \end{aligned} \tag{A.3}$$

As becomes clear from the above expressions, H_N^v cannot be extracted from L_{uv}^{CL} and L_{yv}^{CL}.

Appendix B

QP formulation for constraints handling

QP-formulation of the constraints is well known and available in the literature. Typical process constraints are as follows:

$$\begin{array}{rcccll} u_{min} & \leq & u_t & \leq & u_{max} & \forall t \quad \textit{Amplitude limits} \\ \Delta u_{min} & \leq & \Delta u_t = u_t - u_{t-1} & \leq & \Delta u_{max} & \forall t \quad \textit{Slewrate limits} \\ y_{min} & \leq & y_t & \leq & y_{max} & \forall t \quad \textit{Quality limits} \end{array}$$

These constraints for the predictive controller can be expressed as

$$\begin{array}{rcccll} u_{min} & \leq & u_{t+k} & \leq & u_{max} & k = 0, 1, 2, ..., N_u - 1 \\ \Delta u_{min} & \leq & \Delta u_{t+k} & \leq & \Delta u_{max} & k = 0, 1, 2, ..., N_u - 1 \\ y_{min} & \leq & \hat{y}_{t+k|t} & \leq & y_{max} & k = 1, 2, ..., N_2 \end{array}$$

Define

$$L_1 = \left[\begin{array}{cccc} \Delta u_{min} & ... & ... & \Delta u_{min} \end{array} \right]^T;$$

$$U_1 = \left[\begin{array}{cccc} \Delta u_{max} & ... & ... & \Delta u_{max} \end{array} \right]^T;$$

$$L_2 = \left[\begin{array}{c} u_{min} - u_{t-1} \\ ... \\ ... \\ u_{min} - u_{t-1} \end{array} \right]; \qquad U_2 = \left[\begin{array}{c} u_{max} - u_{t-1} \\ ... \\ ... \\ u_{max} - u_{t-1} \end{array} \right];$$

$$
\begin{aligned}
L_3 &= \begin{bmatrix} y_{min} & \cdots & \cdots & y_{min} \end{bmatrix}^T - F; \\
U_3 &= \begin{bmatrix} y_{max} & \cdots & \cdots & y_{max} \end{bmatrix}^T - F; \qquad \textit{and} \\
\mathcal{R} &= \begin{bmatrix} 1 & 0 & \ldots & 0 \\ 1 & 1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 1 & 1 & 1 & 1 \end{bmatrix}
\end{aligned}
$$

The constraints can be rewritten as:

$$
\begin{aligned}
L_1 &\leq \quad \triangle u_f \quad &&\leq U_1 \\
L_2 &\leq \quad \mathcal{R}\triangle u_f \quad &&\leq U_2 \\
L_3 &\leq S_{N_2,N_u}\triangle u_f &&\leq U_3
\end{aligned}
$$

These constraints can be combined to the form of a single matrix inequality:

$$
\mathcal{A}\triangle u \leq \mathcal{B} \tag{B.1}
$$

with

$$
\begin{aligned}
\mathcal{A} &= \begin{bmatrix} -I & -\mathcal{R} & -S_{N_2,N_u} & I & \mathcal{R} & S_{N_2,N_u} \end{bmatrix}^T \\
\mathcal{B} &= \begin{bmatrix} -L_1 & -L_2 & -L_3 & U_1 & U_2 & U_3 \end{bmatrix}^T
\end{aligned}
$$

Appendix C

Equivalence of subspace and GPC predictor matrices

The vector of predictor equations used in subspace based predictive controller is

$$\begin{bmatrix} \hat{y}_{t+1} \\ \dots \\ \hat{y}_{t+N_2} \end{bmatrix} = L_w(1:N_2m,:) \begin{bmatrix} y_{t-N+1} \\ \dots \\ y_t \\ u_{t-N} \\ \dots \\ u_{t-1} \end{bmatrix} + L_u(1:N_2m,1:N_ul) \begin{bmatrix} u_t \\ \dots \\ u_{t+N_u-1} \end{bmatrix}$$

Therefore for a k-step ahead prediction

$$\hat{y}_{t+k} = L_w((k-1)m+1:km,:) \begin{bmatrix} y_{t-N+1} \\ \dots \\ y_t \\ u_{t-N} \\ \dots \\ u_{t-1} \end{bmatrix} + L_u((k-1)m+1:km,:) \begin{bmatrix} u_t \\ \dots \\ u_{t+N_u-1} \end{bmatrix}$$

$$
\begin{aligned}
&= L_w^o((k-1)m+1:km,:)\begin{bmatrix} y_{t-N+1} \\ \dots \\ y_t \\ u_{t-N} \\ \dots \\ u_{t-1} \end{bmatrix} \\
&\quad +S_{N_2,m}((k-1)m+1:km,:)\begin{bmatrix} \triangle u_t \\ \dots \\ \triangle u_{t+N_u-1} \end{bmatrix} \\
&= \begin{bmatrix} p_{N-1} & \dots & p_1 & p_0 & q_{N-1} & \dots & q_1 & q_0 \end{bmatrix}\begin{bmatrix} y_{t-N+1} \\ \dots \\ y_t \\ u_{t-N} \\ \dots \\ u_{t-1} \end{bmatrix} \\
&\quad +\begin{bmatrix} s_{k-1} & \dots & s_1 & s_0 \end{bmatrix}\begin{bmatrix} \triangle u_t \\ \dots \\ \triangle u_{t+m-1} \end{bmatrix} \\
&= [p_0+p_1z^{-1}+\dots+p_{N-1}z^{-N+1}]y_t \\
&\quad +[q_0+q_1z^{-1}+\dots+q_{N-1}z^{-N+1}]u_{t-1} \\
&\quad +[s_0+s_1z^{-1}+\dots+s_{k-1}z^{-z+1}]\triangle u_{t+k-1} \\
&= P(z^{-1})y_t+Q(z^{-1})u_{t-1}+S_k\triangle u_{t+k-1} \qquad \text{(C.1)}
\end{aligned}
$$

Comparing the above equation with equation (4.5), we observe that

$$P = \frac{F_k}{C}; \qquad Q = \frac{\Gamma_k\triangle}{C}; \qquad S_k = G_k \qquad \text{(C.2)}$$

Therefore, the parametric matrices obtained in GPC design by first identifying an ARIMAX model for the process and then through recursive solution of the Diophantine equations [7], are directly identified by the subspace identification. This removes the requirement of pre specifying order and structure (ARIMAX) for the process model.

Appendix D

Equivalent subspace representation

We can re write equation (5.14)

$$y_{t+1} = l_{y_p} \begin{bmatrix} y_{t-N+1} \\ \dots \\ y_t \end{bmatrix} + l_{u_p} \begin{bmatrix} u_{t-N+1} \\ \dots \\ u_t \end{bmatrix} + e_{t+1} \tag{D.1}$$

We can also express y_{t+1} as

$$\begin{aligned}
y_{t+1} &= y_t + G_1 u_t + L_1 e_t + e_{t+1} \\
&= l_{y_p} \begin{bmatrix} y_{t-N} \\ \dots \\ y_{t-1} \end{bmatrix} + l_{u_p} \begin{bmatrix} u_{t-N} \\ \dots \\ u_{t-1} \end{bmatrix} + G_1 u_t + L_1 e_t + e_{t+1} \\
&= l_{y_p} \begin{bmatrix} y_{t-N-1} \\ \dots \\ y_{t-2} \end{bmatrix} + l_{u_p} \begin{bmatrix} u_{t-N-1} \\ \dots \\ u_{t-2} \end{bmatrix} + \begin{bmatrix} G_1 & G_2 \end{bmatrix} \begin{bmatrix} u_t \\ u_{t-1} \end{bmatrix} + \begin{bmatrix} L_1 & L_2 \end{bmatrix} \begin{bmatrix} e_t \\ e_{t-1} \end{bmatrix} + e_{t+1} \\
&= \dots. \\
&= l_{y_p} \begin{bmatrix} y_{t-2N+1} \\ \dots \\ y_{t-N} \end{bmatrix} + l_{u_p} \begin{bmatrix} u_{t-2N+1} \\ \dots \\ u_{t-N} \end{bmatrix}
\end{aligned}$$

$$+\left[\begin{array}{ccc} G_1 & \ldots & G_N \end{array}\right]\left[\begin{array}{c} u_t \\ \ldots \\ u_{t-N+1} \end{array}\right]+\left[\begin{array}{ccc} L_1 & \ldots & L_N \end{array}\right]\left[\begin{array}{c} e_t \\ \ldots \\ e_{t-N+1} \end{array}\right]+e_{t+1}$$

As $N \longrightarrow \infty$, an equivalent expression of y_{t+1} in terms of the past inputs and the past noise can be written as

$$y_{t+1} = \left[\begin{array}{ccc} G_1 & \ldots & G_N \end{array}\right]\left[\begin{array}{c} u_t \\ \ldots \\ u_{t-N+1} \end{array}\right]+\left[\begin{array}{ccc} L_1 & \ldots & L_N \end{array}\right]\left[\begin{array}{c} e_t \\ \ldots \\ e_{t-N+1} \end{array}\right]+e_{t+1} \quad \text{(D.2)}$$

Appendix E

Simplified matrix expressions from singular value decomposition

Let A represent an $m \times n$ matrix of rank $r \leq min(m,n)$. The so that

$$A = X\Sigma Y^T = \begin{bmatrix} X_1 & X_2 \end{bmatrix} \begin{bmatrix} \Sigma_r & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} Y_1^T \\ Y_2^T \end{bmatrix} = X_1 \ \Sigma_r \ Y_1^T \tag{E.1}$$

where r is the number of singular values of A that are not zero. The matrices X and Y are orthogonal matrices i.e., the columns are mutually orthogonal vectors of unit length and are non-unique. We can derive

$$A^T A = (Y_1 \ \Sigma_r \ X_1^T) \ (X_1 \ \Sigma_r \ Y_1^T) = Y_1 \ \Sigma_r^2 \ Y_1^T \tag{E.2}$$

$$(A^T A)^\dagger = (Y_1 \ \Sigma_r^2 \ Y_1^T)^\dagger = Y_1 \ \Sigma_r^{-2} \ Y_1^T \tag{E.3}$$

Therefore we have

$$A^\dagger A = (A^T A)^\dagger A^T A = \begin{bmatrix} Y_1 \ \Sigma_r^{-2} \ Y_1^T \end{bmatrix} \begin{bmatrix} Y_1 \ \Sigma_r^2 \ Y_1^T \end{bmatrix} = Y_1 \ Y_1^T \tag{E.4}$$

$$AA^\dagger = A(A^T A)^\dagger A^T = \begin{bmatrix} X_1 \ \Sigma_r \ Y_1^T \end{bmatrix} \begin{bmatrix} Y_1 \ \Sigma_r^{-2} \ Y_1^T \end{bmatrix} \begin{bmatrix} Y_1 \ \Sigma_r \ X_1^T \end{bmatrix} = X_1 \ X_1^T \tag{E.5}$$

The matrices X and Y are unitary matrices. Hence we can also write [39]

$$I - A^\dagger A = Y_2 \ Y_2^T \tag{E.6}$$

$$I - AA^\dagger = X_2 \ X_2^T \tag{E.7}$$

Both the matrices $(I - A^{\dagger}A)$ and $(I - AA^{\dagger})$ are idempotent. Note that Y_2 is an $n \times (n-r)$ matrix. Hence $I - A^{\dagger}A$ will have a rank of $n - r$.

Appendix F

Essentially disjoint condition

From [39]

Lemma 17.1.4. Let U and V represent subspaces of $R^{m\times n}$, then

(1) U and V are essentially disjoint if and only if, for matrices $\mathbf{U}\in U$ and $\mathbf{V}\in V$, the only solution to the matrix equation

$$\mathbf{U}+\mathbf{V} = 0 \tag{F.1}$$

is $\mathbf{U}=\mathbf{V}=0$; and

(2) U and V are essentially disjoint if and only if, for every non-null matrix $\mathbf{U}\in U$ and every non-null matrix $\mathbf{V}\in V$, $\mathbf{U}$ and $\mathbf{V}$ are linearly independent.

We assumed that the process transfer function $G(z^{-1})$ is equation (7.34) is full rank with proper and stable transfer functions. Hence the following claims can be made

(a) The vectors $\begin{bmatrix} G_0 \\ G_1 \\ G_2 \\ \cdots \end{bmatrix}, \begin{bmatrix} 0 \\ G_0 \\ G_1 \\ \cdots \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ G_0 \\ \cdots \end{bmatrix}, \ldots$ are all essentially disjoint to each other.

(b) The matrices $L_u = \begin{bmatrix} G_0 & 0 & \dots & 0 \\ G_1 & G_0 & \dots & 0 \\ G_2 & G_1 & \dots & 0 \\ \dots & \dots & \dots & \dots \end{bmatrix}$ and $\begin{bmatrix} G_1 \\ G_2 \\ G_3 \\ \dots \end{bmatrix}$ are essentially disjoint.

Appendix G

Corollaries from [39]

(1) Corollary 17.5.2 Let $\mathcal{A}$ represent an $m \times n$ matrix and $\mathcal{B}$ an $n \times p$ matrix. Then,

$$rank(\mathcal{AB}) = rank(\mathcal{A}) + rank(\mathcal{B}) - n + rank\left[(I - \mathcal{BB}^{\dagger})(I - \mathcal{A}^{\dagger}\mathcal{A})\right] \quad \text{(G.1)}$$

(2) Corollary 17.2.10 Let $\mathcal{A}$ represent an $m \times n$ matrix, $\mathcal{B}$ an $m \times p$ matrix. Then rank$\left[(I - \mathcal{AA}^{\dagger})\mathcal{B}\right]$ = rank $(\mathcal{B})$ if and only if C$(\mathcal{A})$ and C$(\mathcal{B})$ are essentially disjoint.

(3) From chapter 18 Let $\mathcal{R}$ represent an $n \times q$ matrix, $\mathcal{S}$ an $n \times m$ matrix, $\mathcal{T}$ an $m \times p$ matrix, and $\mathcal{U}$ a $p \times q$ matrix. Then,

$$\begin{aligned} rank(\mathcal{R} + \mathcal{STU}) = {} & rank(\mathcal{R}) + rank(\mathcal{Q}) + rank(\mathcal{M}) + rank(\mathcal{N}) \\ & + rank\left[(I - \mathcal{MM}^{\dagger})\mathcal{X}\mathcal{Q}^{\dagger}\mathcal{Y}(I - \mathcal{N}^{\dagger}\mathcal{N})\right] \\ & - rank(\mathcal{T}) \end{aligned} \quad \text{(G.2)}$$

where

$$\begin{aligned} \mathcal{E}_{\mathcal{R}} &= I - \mathcal{RR}^{\dagger};\ \mathcal{F}_{\mathcal{R}} = I - \mathcal{R}^{\dagger}\mathcal{R};\ \mathcal{X} = \mathcal{E}_{\mathcal{R}}\mathcal{ST};\ \mathcal{Y} = \mathcal{TUF}_{\mathcal{R}}; \\ \mathcal{M} &= \mathcal{X}(I - \mathcal{Q}^{\dagger}\mathcal{Q});\ \mathcal{N} = (I - \mathcal{QQ}^{\dagger})\mathcal{Y} \end{aligned}$$

Refer to [39] for proofs of the above corollaries.

Appendix H

Scope for future work

The work treated in this thesis is derived from innovations form representation for multivariate linear systems and the subspace matrices are of the N4SID type obtained with least squares solution. Several subspace methodologies are available in the literature which use different numerical techniques such as CVA and PCA in their identification algorithms. It would be interesting to use subspace matrices obtained by different numerical techniques to extend the work presented in this thesis. More over, with appropriate modifications to the representation of the system, the work in this thesis can be extended to the case of non-linear systems, time-varying systems, and linear systems with special data and process conditions such as the presence of state noise, process input noise and correlated measured and/or unmeasured disturbances to the process.

Interactor matrices are used in some areas of process control apart from using it to obtain the multivariate MVC-benchmark as shown by [36, 42, 46]. The interactor free approach to control presented in chapter 7 has applications in such areas as multivariate recursive/adaptive control.

Time delays are considered as infinite zeros. For non-minimum phase (NMP) systems, the interactor should also include the NMP zeros. As shown in [42, 46], the interactor used in the NMP case is obtained by multiplying the interactor matrix

corresponding to the time delays (infinite zeros), D_{inf}, with the interactor matrix corresponding to the NMP zeros, D_{NMP}. The interactor free multivariate MVC-benchmark methodology in chapter 7 is applicable for only minimum phase systems. For non-minimum phase systems the interactor $(I - LuLu^{\dagger})$ needs to be modified to account for the NMP zeros. This remains an issue open for research.

Fault detection and identification (FDI) is an emerging area of control research which, apart from detecting sensor faults and actuator faults, widely makes use of subspace methodologies to detect 'parametric' changes to the real systems over time. FDI methodologies typically involve recursively identifying the interested parameters and checking for any changes with statistical tests. In some FDI methodologies state space representation is used and the computations are quite involved for practical application on real systems. Since subspace representation is equivalent to state space representation, it would be interesting to verify if formulation of the FDI problem as a function of subspace matrices can simplify the computations involved.

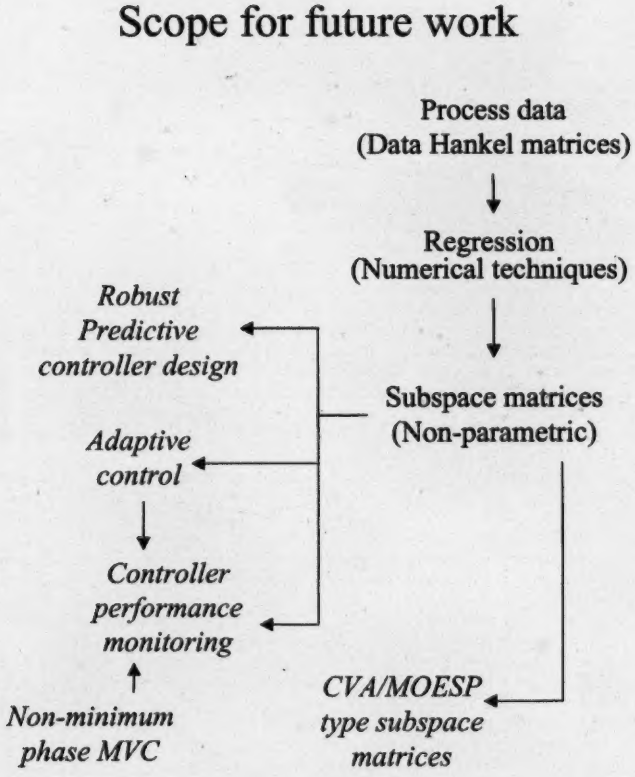


Figure H.1: Future research scope